YOUR OFFICIAL RASPBERRY PI MAGAZINE

The MagPi

Issue 144 | August 2024 | magpi.cc

The official Raspberry Pi magazine

ESSENTIAL ELECTRONICS

GET SMART WITH CIRCUIT PROGRAMMING

- Private Cloud Server
- Learn to use AI Kit
- Twirly Shirley turntable

£5.99
08
9 772051 998001
PRESS

INCREDIBLE VISION AND PHOTO PROJECTS

Industrial Raspberry Pi
ComfilePi

UL LISTED

Powered by Raspberry Pi

The ComfilePi is a touch panel PC designed with high-tolerant components and no moving parts for industrial applications. It features a water-resistant front panel, touchscreen, color LCD (available in various sizes), RS-232, RS-485, Ethernet, USB, I2C, SPI, digital IO, battery-backed RTC (real-time clock), and piezo buzzer.

Use the rear-panel 40-pin GPIO header to expand its features and capabilities with additional I/O boards. The ComfilePi is UL Listed and employs Raspberry Pi Compute Module.

Visit www.comfiletech.com

COMFILE TECHNOLOGY

WELCOME
to The MagPi 144

Attaching a camera to Raspberry Pi opens up a world of vision-based possibilities. I can take photographs, but now thanks to AI Kit my Raspberry Pi can make smart decisions based on what it sees.

I've been having a lot of fun this month with Raspberry Pi's Camera Module and AI Kit. Toby, our maker-in-residence, glued a GorillaPod to the bottom of an official Raspberry Pi case and made a portable smart camera to take out for testing. My three-legged camera and AI Kit creation has yet to get a name (PiTriPod, maybe?)

Read all about how to set up AI Kit on Page 46, Camera on Page 56, and amazing camera projects to try out on Page 62.

Of course, what makes Raspberry Pi special isn't just the board but what you can make with it (just about anything, really!) Our Essential Electronics guide on Page 32 has everything you need to start making your own circuit creations with Raspberry Pi.

Nothing beats building a good project.

Lucy Hattersley Editor

EDITOR

Lucy Hattersley

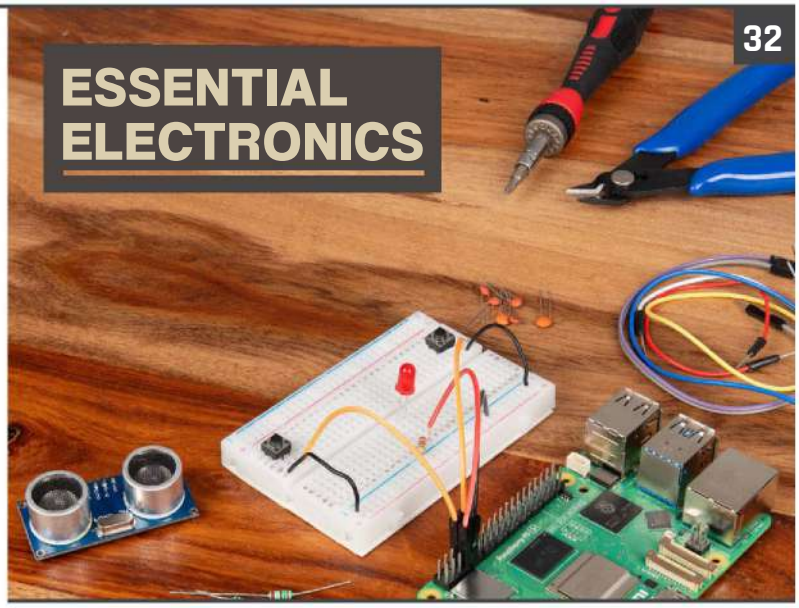
Lucy is editor of The MagPi and is training her camera to detect a puppy to adopt. Don't tell her pet tripod project.

magpi.cc

Contents

Issue 144 > August 2024

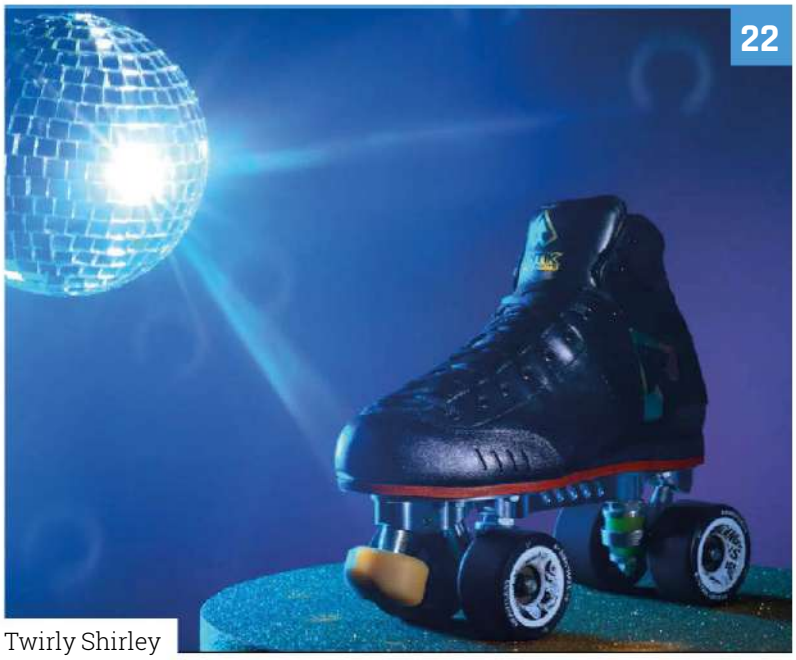
Cover Feature



Regulars

Project Showcases

Andotrope



Twirly Shirley

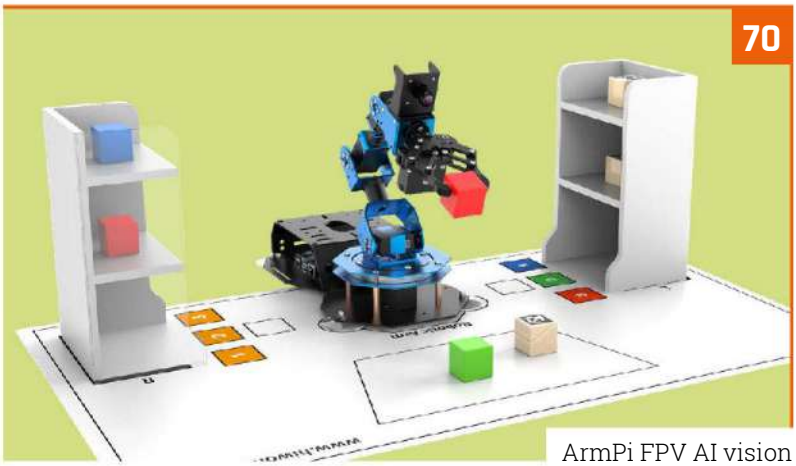
The MagPi is published monthly by Raspberry Pi Ltd, 194 Cambridge Science Park, Milton Road, Cambridge, England, CB4 0AB. Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701, is the mailing agent for copies distributed in the US and Canada. Application to mail at Periodicals prices is pending at Williamsport, PA. POSTMASTER: Send address changes to The MagPi, c/o Publishers Service Associates, 2406 Reach Road, Williamsport, PA, 17701.

Tutorials

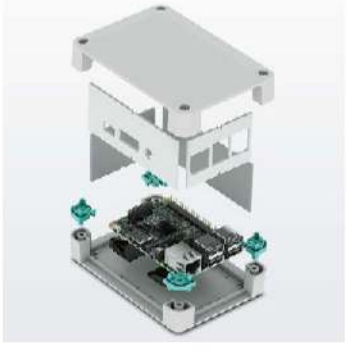
The Big Feature

Reviews

Community



WIN HOME ASSISTANT **YELLOW** 94



Raspberry Pi finds a new home

Versatile enclosure for Raspberry Pi B+

The new UCS Universal Case System is now available with ready made cut-outs for the 7" touch display and standard connections of the Raspberry Pi B+ single board computers. The 237 x 195 x 47mm housings are available in black or grey and are suitable for wall or desktop mounting

For additional information call 01952 681700 or visit

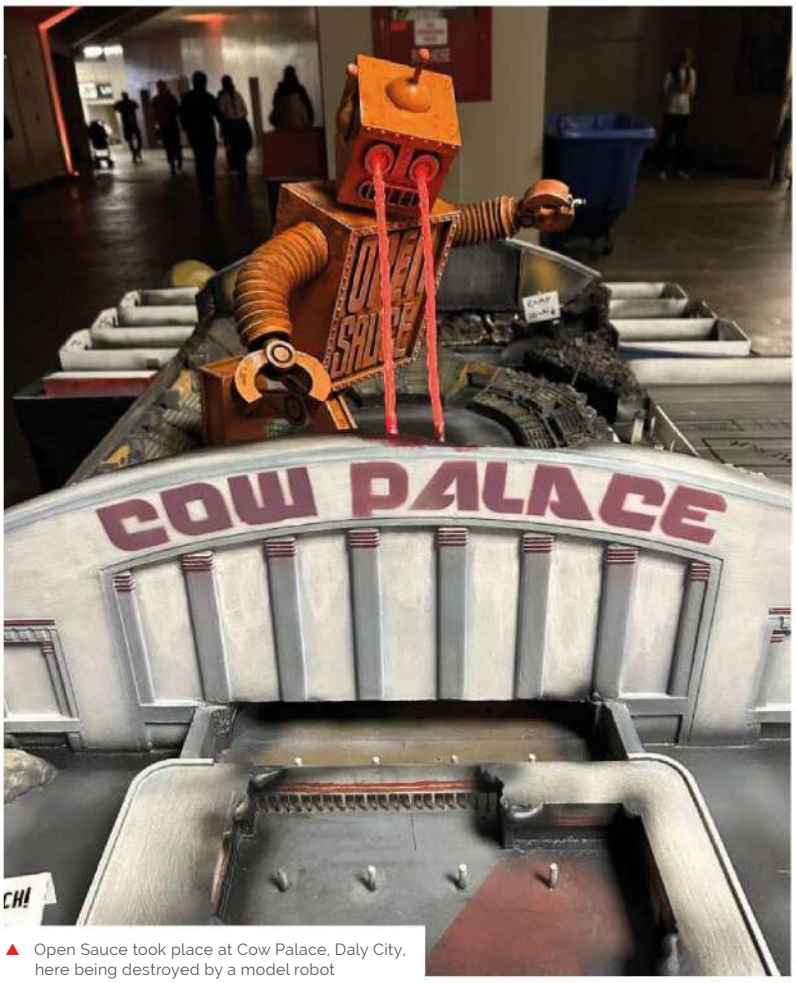
https://phoe.co/ucs-series

PHOENIX CONTACT
INSPIRING INNOVATIONS

Raspberry Pi at Open Sauce 2024

We had quite the weekend with makers in San Francisco.

By **Ashley Whittaker**



▲ Open Sauce took place at Cow Palace, Daly City, here being destroyed by a model robot

▲ Kevin McAleer had many creations to demonstrate

We're just back from Open Sauce 2024 (opensauce.com) and I don't think we've ever seen so many makers, tech enthusiasts, and YouTubers all together in one place before. This year's Open Sauce expanded into the Cow Palace venue in San Francisco and definitely felt more gigantic than last year's inaugural event. Our poor feet; please send help for our feet.

The event combines the best of makers and creators, featuring interactive exhibits, an up-close look at maker technology, talks, panels, and small workshops with top YouTubers.

> Thanks again to everyone who stopped to talk to us about your creations

Kev's Voltara Fortune Teller prints fortunes of varying accuracy at the push of a button

Pi-powered creations everywhere

It was excellent to meet so many makers who had brought their Raspberry Pi-powered creations to Open Sauce. Over the coming weeks, we'll be blogging about as many of them as we can: there's a penny press, an ocular impairment assessment tool, a Connect 4 clock, and a photo booth inside a vintage camera.

We were too busy perusing the maker booths to get to the panels, but we did set eyes on a handful of YouTubers as they popped out to meet people in the main halls. It's an incomplete list, but we spotted Mark Rober, Adam Savage, The Hacksmith, Ruth Amos, Estefannie, and Becky Stern, whose pet cat and mermaid hair we enthused about recently.

Special shout-out to Tex Kang's (not Pi-powered but still cool) AI tote bag printing station. We asked it to print us a raspberry pie PCB, and it did a pretty good job.

This PCB decorated like a raspberry pie (or possibly a tart) was printed by an AI-powered system

PIZ dispensers and fortune tellers

Jeff Geerling gave away 480 Raspberry Pi Picos at the end (that's an entire reel!) and he didn't just hand them out like a civilian. He (of course) over-engineered an eye-catching alternative.

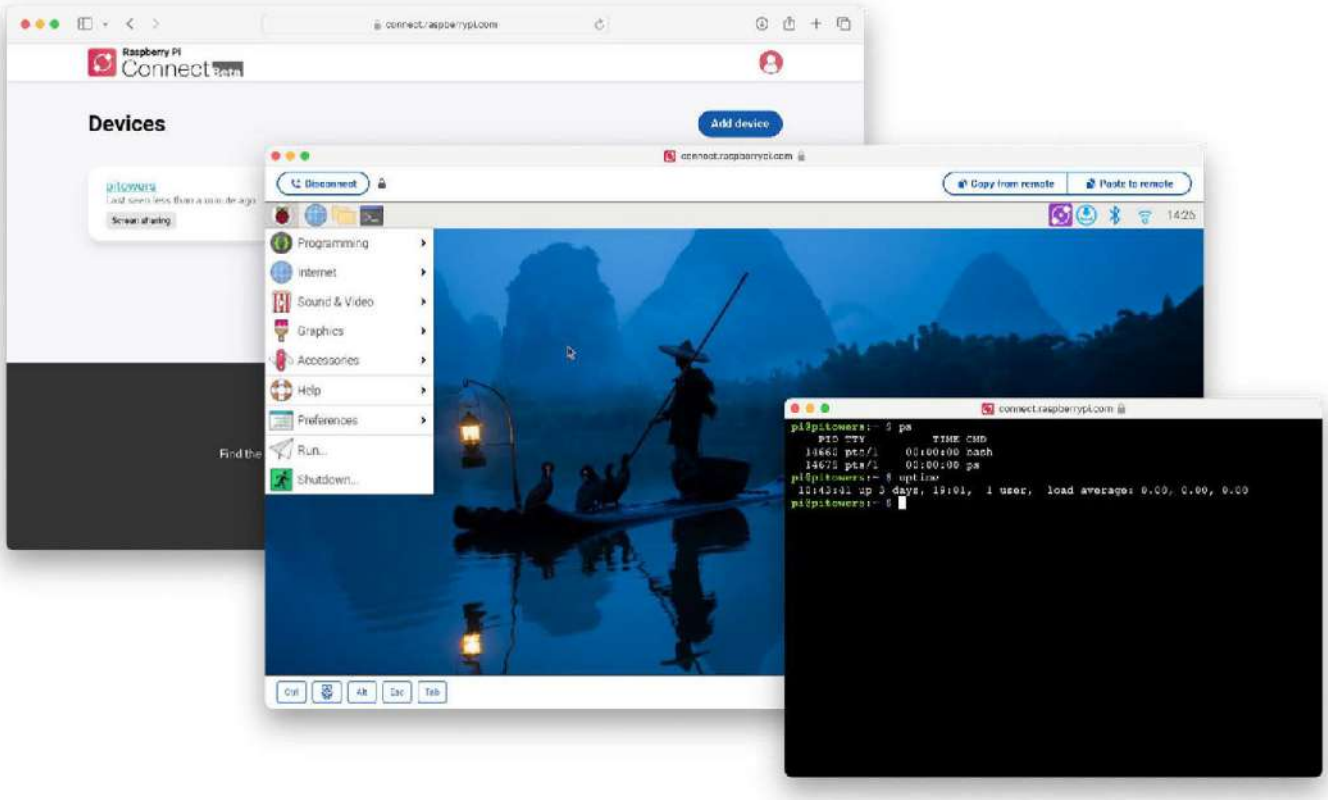
Jeff went for a little nostalgia hit, copying the iconic PEZ dispenser design but blowing up the size so it was big enough to hold loads of Picos rather than those impossibly tiny candy capsules. We didn't actually manage to cross paths with Jeff this year, so vast was the venue, but we did see his PIZ dispenser taking a rest at Kevin McAleer's (**magpi.cc/kevinmcaleer**) booth. There we also had our fortune read by a robot named Voltara, inspired by the Zoltar machine from the movie Big.

Kev made Voltara (**magpi.cc/kevfortune**) especially for Open Sauce. It's powered by a Raspberry Pi 4 and features a thermal printer and arcade-style buttons.

Thanks again to everyone who stopped to talk to us about your creations. We were overwhelmed by the number of Pis in the wild we spotted. We can't wait to see you again next year.

Raspberry Pi
Connect updated

Remote shell access and support for older devices.
By **Chris Lowder**



▲ Raspberry Pi Connect enables you to remotely access Raspberry Pi OS, and now works with all Raspberry Pi computers

A month and a half ago, we launched Raspberry Pi Connect (magpi.cc/connect), giving you simple, remote access to your Raspberry Pi straight out of the box, from anywhere in the world. Since then we've been listening to your thoughts on what you'd most like to see from the service: today we're excited to announce the latest beta release, bringing Raspberry Pi Connect to even more devices.

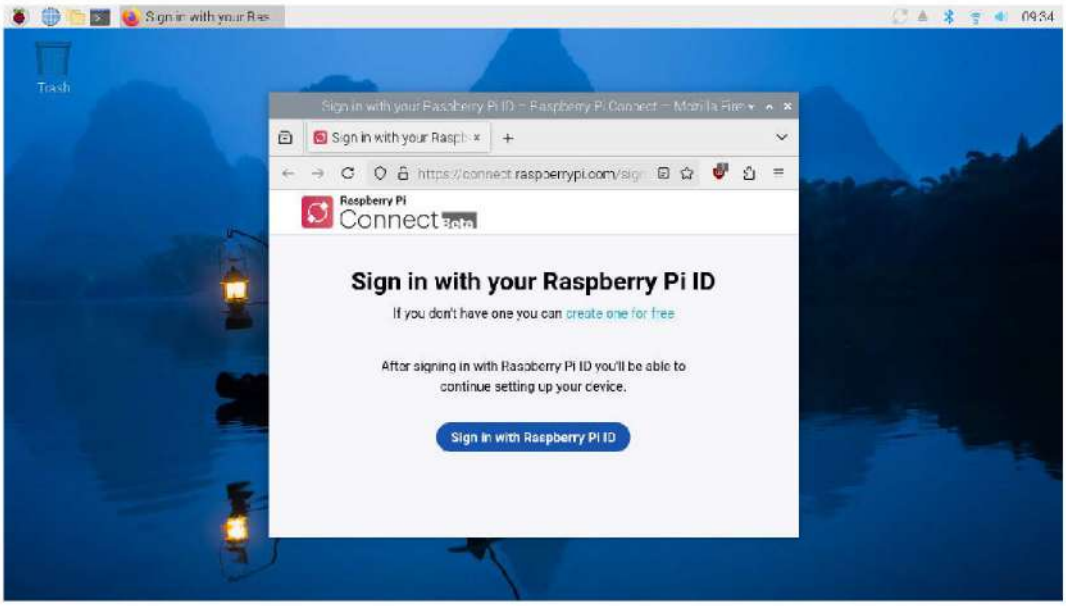
Today's release includes remote shell access and support for all Raspberry Pi computers, whether they're running Raspberry Pi OS 32-bit or 64-bit, going all the way back to the 2012 Raspberry Pi 1. This includes Raspberry Pi OS Lite, as well as versions of Raspberry Pi OS with the desktop.

Remote shell access

The new remote shell feature in Raspberry Pi Connect enables you to launch a shell on your Raspberry Pi device from a web browser, over a secure connection.

This provides access to your Raspberry Pi without a desktop environment, extending support to older devices as well as devices running Raspberry Pi OS Lite. Remote shell access also works much better over low-bandwidth connections than screen sharing, making it a handy option to have.

To get started with remote shell on Raspberry Pi OS Lite, run the following commands:



Login to Raspberry Pi Connect with your Raspberry Pi ID

```
sudo apt update
sudo apt install rpi-connect-lite
```

Then, sign in using the rpi-connect command line interface:

```
rpi-connect-signin
```

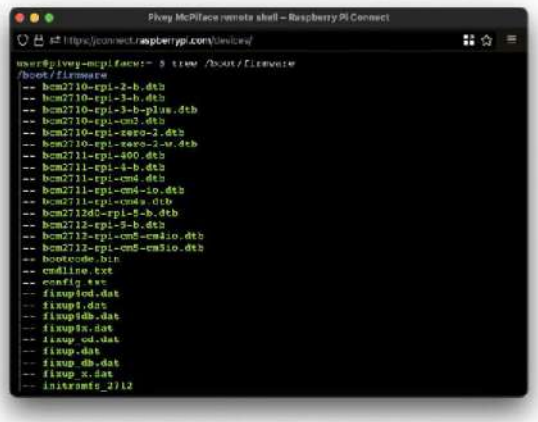
Visit the verification URL on any device and sign in to link your Raspberry Pi with your Raspberry Pi ID.

You also get expanded support for Raspberry Pi devices. Starting with this latest release, version 1.2, Raspberry Pi Connect now supports all 32-bit Raspberry Pi computers, as well as 64-bit computers. All devices get remote shell out-of-the-box, and if you use a Wayland compositor, such as Wayfire, you can also share your screen. In practice, this means you can use screen sharing with Raspberry Pi 4 and later models, and remote shell with all models of Raspberry Pi, even the oldest.

We hope this will make it a little bit easier for people to keep older Raspberry Pi computers in service doing useful stuff. Plenty of remote headless applications, for example, don't need the performance of Raspberry Pi 4 and 5, but still benefit from straightforward remote access.

We're looking forward to seeing how people use these new features in their projects. If you haven't dipped your toe into the water yet, our Raspberry Pi Connect documentation (**magpi.cc/connectdocs**) has everything you need to get started.

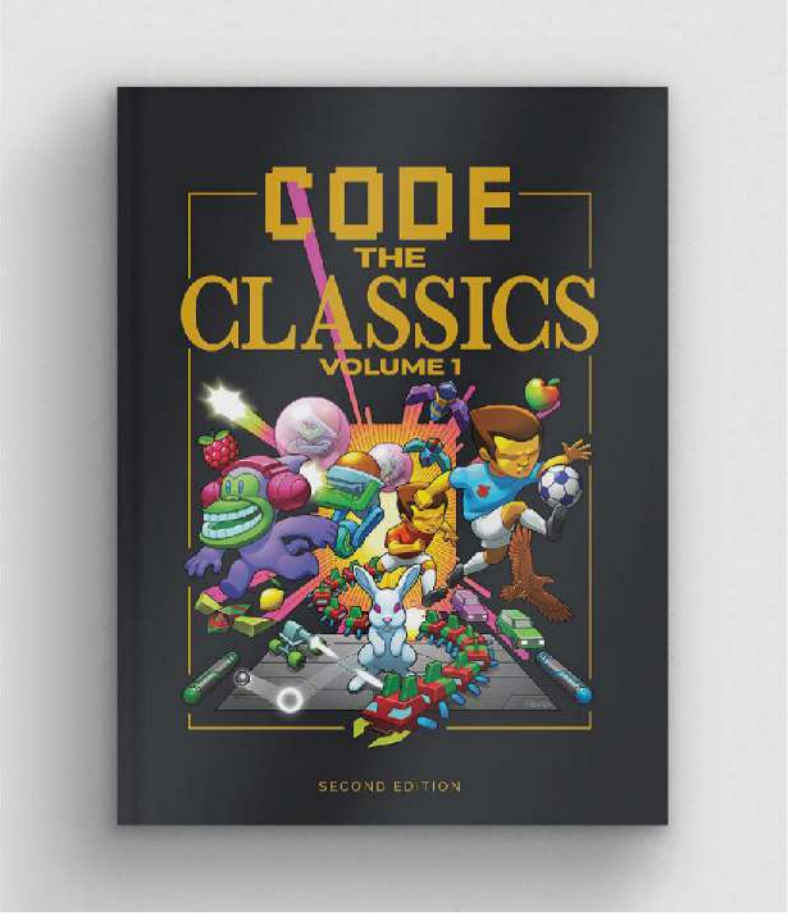
> We hope this will make it a little bit easier for people to keep older Raspberry Pi computers in service



Connect now features remote shell access

New edition of **Code the Classics**

An updated edition of the game programming guide with new chapters is on sale now. By **Brian Jepson**



Code the Classics Volume 1, Second Edition is now available

Our retro-gaming classic went on sale back in 2019, and this year we're releasing the new edition of Code the Classics Volume I. The biggest update is the addition of three new tutorials covering Python, Pygame Zero, and Git for version control. These new tutorials will make working with the code examples easier for beginners.

Retrogamers of certain generations will remember the days of type-in computer game listings. Nearly every computer magazine, from *ANALOG Computing* (**magpi.cc/analogcomp**) to *The Micro User* (**magpi.cc/microuser**) featured code listings that you'd have to type into your computer. There were books, too, like the ubiquitous Usborne books (**magpi.cc/usborne**). You'd trade an hour or two of your time for an experience that was just as good as games from your local computer store.

This new edition of *Code the Classics Volume I* features retro arcade games written by Raspberry Pi co-founder and CEO Eben Upton, ably assisted by Andrew Gillett and Sean Tracey. Dan Malone (famous for his work with The Bitmap Brothers, **magpi.cc/danmalone**) created the game graphics, and long-time game audio pro Allister Brimble (**allisterbrimble.com**) provided the music and sound effects. David Crookes and Liz Upton wrote the stories that take you behind the scenes of the creation of the five classic arcade games featured in this book.

You won't have to type in these games, though. We've set up a GitHub repo with all the source code (**magpi.cc/codeclassicsgit**). However, we've printed abridged listings in the book to approximate the type-in experience.

The simple act of typing in a game helped to familiarise you with a programming language (such as BASIC). By making mistakes, you could

> The simple act of typing in a game helped to familiarise you with a programming language

start to understand what other, more intentional changes might accomplish. Some of Eben's earliest games started off as heavily modified versions of type-in listings. In fact, one of these made a sneaky reappearance (**magpi.cc/ebenbbc**) on this blog back in 2018.

In this new edition, you'll meet these vintage-inspired games, and learn from their code in between rounds of play:

▲ Recreate classic arcade and video games in Python code

▲ There's no finer way to learn to code than typing out a program from a book

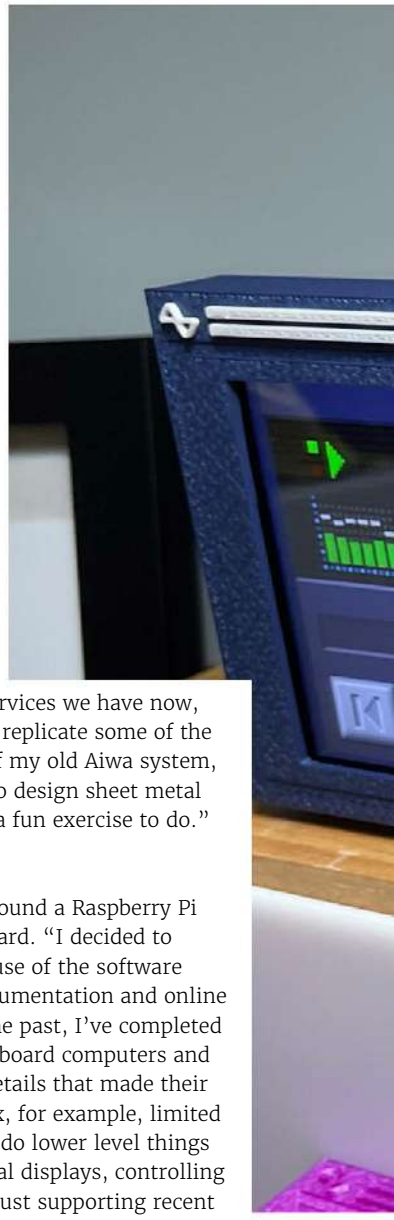
▶ Can you believe they did *Machine Code for Beginners*?

- *Boing!*: all it took was a couple of lines and a dot, and gamers would be queuing up to play.
- *Cavern*: Enduringly popular, the platform game genre is still packed with creative possibilities.
- *Infinite Bunner*: Play around with the benefits that a top-down perspective can lend to the classic platform genre.
- *Myriapod*: Some shooters confine the gameplay to a single screen while limiting the player's movement. Restrictions can build challenge and difficulty, making for truly addictive gaming.
- *Substitute Soccer*: Top-down games of pinball-style soccer kicked off a sports genre that's still going strong.

Stay tuned, because we've been hard at work on the second volume of *Code the Classics*. *Volume II* will feature all-new games and interviews. Get your copy today: **magpi.cc/codetheclassics**!

Linamp

Adore music from the 1980s and 1990s? Then wind back time and play your tracks on this Winamp/hi-fi-inspired set-up. **David Crookes** gets into the groove



MAKER

Rodrigo Méndez

Rodrigo is a software engineer, hardware hobbyist and industrial design aficionado from Mexico. He's been tinkering with hardware since 2010.

magpi.cc/linamp

When looking to play MP3 music files for a good while from the late-'90s onwards, Microsoft Windows users would tend to migrate towards the Winamp media player. Originally released in 1997 and developed by Justin Frankel and Dmitry Boldyrev, Winamp's skeuomorphic user interface soon resembled a compact stereo system. It was instantly accessible and it certainly caught the attention of a young Rodrigo Méndez – so much so, he's based a cool Raspberry Pi project on it.

"I remember when I first got my own PC – an 800Mhz Athlon with Windows 98," he recalls. "My cousin installed Winamp in it and gave me some MP3s to play. At that time, seeing music coming from a PC, and having the visualisations move in real time to the music was unbelievable to me because my previous experience with computers had been an old MS-DOS PC that my dad had at work."

For his project, Rodrigo wanted to bring Winamp's early days back to life in a more physical form so he decided to produce a music player that could easily be part of a hi-fi mini system. Specifically, he took inspiration from a Aiwa mini setup from around 1983, figuring he could create a case and have all of the functions available on a front touchscreen display.

One thing he noticed was the simplicity of design of the old devices. "Old electronics, especially audio devices, managed to look high-tech and yet elegant," he explains. "They managed to fit in well with people's home décor while using relatively simple construction techniques such as sheet metal or wood. I thought that, with all the new manufacturing services we have now, It wouldn't be too hard to replicate some of the construction techniques of my old Aiwa system, so I set out to learn how to design sheet metal parts, which was in itself a fun exercise to do."

Physical

Rodrigo based his build around a Raspberry Pi 4B with a 32GB microSD card. "I decided to use the Raspberry Pi because of the software support and extensive documentation and online resources," he said. "In the past, I've completed projects with other single board computers and I've always found small details that made their usage much more complex, for example, limited documentation on how to do lower level things like interfacing with special displays, controlling the boot process, or even just supporting recent Linux kernels.

"Since this was a hobby project, I didn't want to spend too much time figuring out those details on other SBCs. Also I chose Raspberry Pi 4 because it was the latest model at the time I started the project. Raspberry Pi 5 was announced after I'd begun and I figured I didn't really need that device's performance anyway."

At first, Rodrigo toyed with the idea of making the front face physical. "I wanted to add real-life buttons using a 3D printer and I was going to design a custom PCB for the interface," he says. But he couldn't find screens with the correct size and dimensions for the spectrum analyser and song information displays that needed to be included to remain faithful to Winamp without making the build too complex.

The intention was always to create a standalone appliance. The casing was designed with a square hole so that a custom back panel could be created to host the connections

Rodrigo had considered a front cover 3D printed on Nylon but he opted for a multi-colour 3D print created on his Bambulab P1S 3D printer instead

The touchscreen display has been based on Winamp 2.0's Base Skin. This version was released in 1998 – two years before Rodrigo got his own PC

Quick FACTS

- It's a music player inspired by the Winamp app
- All of the functions are on a touchscreen
- The software has been written from scratch
- The case design is based on retro hi-fi systems
- It cost around $700 to make

This is the Aiwa stereo tuner and amplifier which inspired Rodrigo to create his own retro hi-fi device

▲ A Raspberry Pi 4 forms the heart of this build. It is connected to a generic 7.9-inch touchscreen display. Extension cables allow for rear ports

"Making it physical would have made it less flexible as well," he adds. For example, I would not be able to have the playlist view in the same display, which would make the device bigger and clunkier so I decided to go with a touchscreen." This, he figured, would stretch across the entire front of the device and include all of the buttons as part of the UI. He reckoned his background as a software engineer meant this would prove to be relatively straightforward.

A little more love

Rodrigo opted for a 7.9-inch extra-wide screen and he created the case using sheet metal, sending his Onshape designs to PCBWay so they could be produced using 1mm anodised aluminium. He'd closely analysed his Aiwa system so he could figure the correct dimensions and overall look and, for convenience, he placed a power button, Ethernet port, USB-A and USB-ports and 3.5mm stereo jack at the back.

The majority of his time was spent on the software, however. He created a custom Qt 6 app which he wrote in C++. "Replicating the layout itself wasn't that hard, it was just a matter of getting rough dimensions at first and drafting the UI in Qt Creator using the graphical Qt

▶ Rodrigo is looking to create Linamp devices for sale and wants the community to help him figure the logistics by filling in survey at **magpi.cc/linampsurvey**

▲ Linmp can play MP3, M4A, FLAC files and more. You can control the output volume and balance too

Widget designer interface, and then fine tuning dimensions by comparing screenshots in GIMP," he explains. "The complex part was replicating the spectrum analyser and getting the raw audio data to feed it. I did a lot of research and looked into other open-source projects' code to get ideas and code to base my solution on."

It's proven to be a very vivid, eye-catching solution and it doesn't feel as if any compromises have been made. Above all, it has the intended nostalgic flavour thanks in the main to the spectrum analyser and the way the name of the song and artist scroll across the top. It also retains the intuitive UI in full, so songs can be selected and played in no time. "Linamp supports playing

> "I looked into other open-source projects' code to get ideas and code"

music CDs from a CD/DVD drive connected via USB as well, but you need to use a powered USB hub in order for the drive to work, otherwise Raspberry Pi may not have enough power for the external drive," Rodrigo adds.

Yet despite being feature packed, Rodrigo wants more. He's also been overwhelmed by the positive reaction from people who have seen the project and he is working on adding Spotify playback and Bluetooth. "It had an amazing response," Rodrigo says. "I never imagined it would have been that popular, especially on YouTube, where I uploaded a quick video of how the UI works as an afterthought." With more than 100,000 views, we don't think we've heard the last of this project.

Pump up the volume

01 First of all, press the power button on the back of the device to turn it on. The screen will remain blank while Raspberry Pi boots directly into the Linamp UI. Once done, it's ready for the selection and playing of music.

02 Touch the PL or Eject button in the UI and you'll go to the Playlist view. A file browser lets you navigate the Raspberry Pi filesystem, starting at the **~/Music** folder by default. Songs can be stored on a home server or USB drive.

03 Select music files and click a button to add them to the playlist then start playing your fabulous tracks. The track information is displayed along with the bitrate and sample rate and there's a real-time bar spectrum analyser.

Gehn Imager Andotrope

Replicating a beloved display from a favourite game yielded amazing results for one Raspberry Pi maker, discovers **Rosie Hattersley**

MAKER

Mike Ando

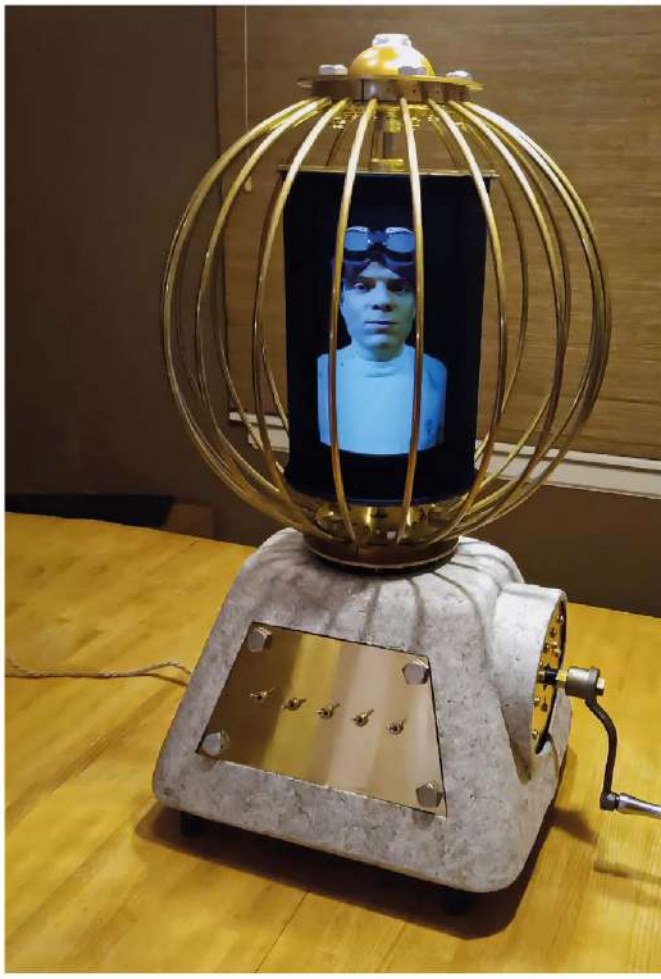
DeLorean driver Mike looks after IT for a small business and says others describe him as a time-travelling mad scientist

magpi.cc/andotrope

Mike Ando loves making things in his spare time, and uses a Raspberry Pi "because they're so easy and save me time implementing things". Even so, he's just spent eight months designing his own take on Gehn's holographic imager from the game *Riven*, the sequel to puzzle adventure *Myst*. As well as displaying images and playing video clips, it can be used for video calls and games. "Basically, I've brought a 150-year-old children's toy into the 21st century." Actually, the Australian systems administrator has created two such 'andotropes'. The second version uses Pico W which is much smaller and cheaper and doesn't use so much expensive and heavy brass, but The MagPi was especially taken with his original, Raspberry Pi 3 model which requires no CGI or other visual trickery and uses two back to back 10in tablets: **magpi.cc/andotropeyt**.

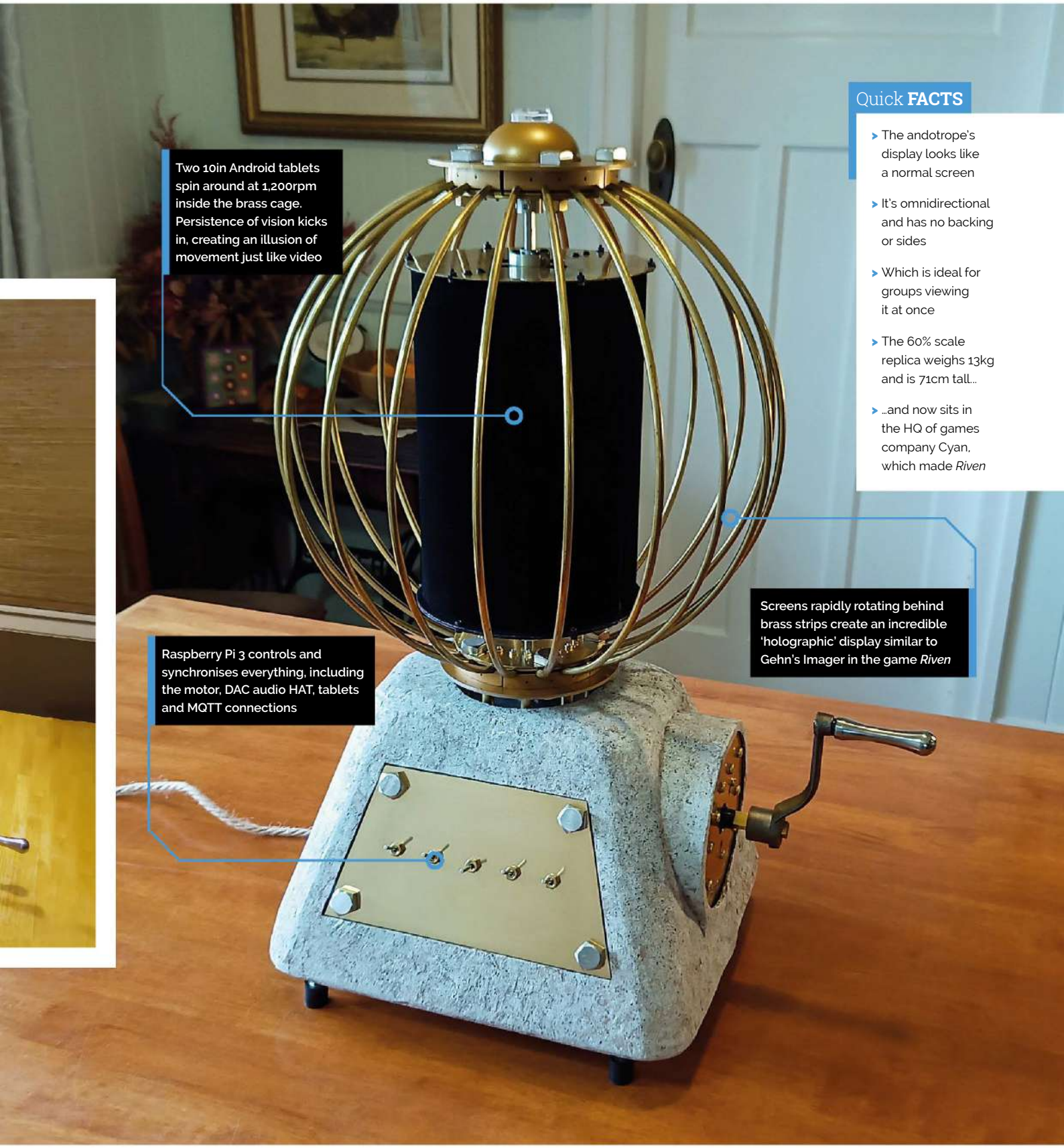
New tricks

If 'andotrope' sounds unfamiliar, that's because it is a word Mike coined, having almost by chance ended up creating a new form of display. His hologram-like display sits inside a spherical cage constructed from around 12m (40ft) of 6mm brass tube. "It doesn't quite meet the definition of a zoetrope, and because I couldn't find anyone else who successfully pulled this off before, I'm calling any display you could build that's similar to this by the generic term 'andotrope'".



▶ Mike managed to display his own image in the andotrope

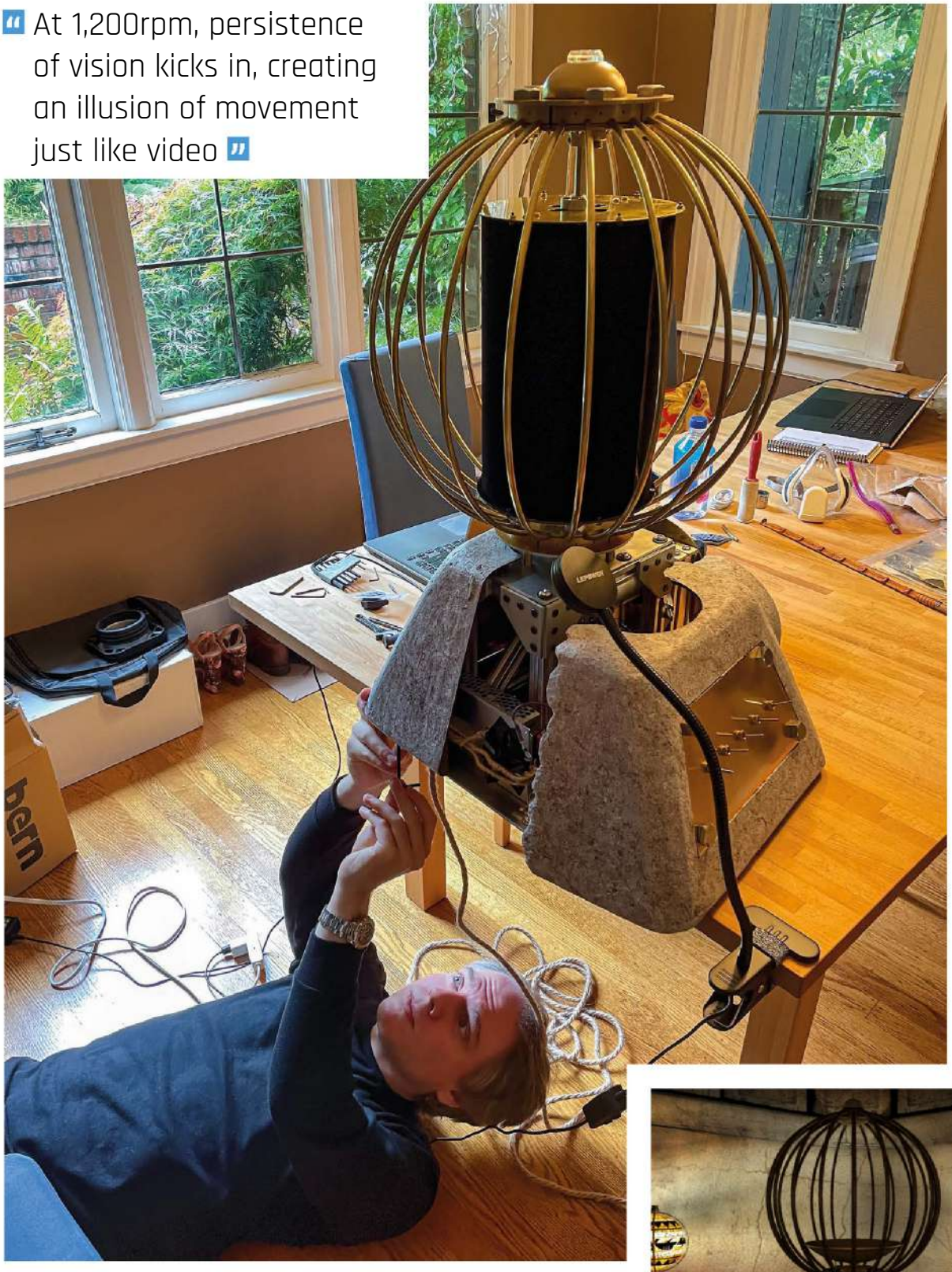
The whole imager is actually 60% scale of the original in-game one. "The original was just impractically large and everyone who has seen it in-person (including the game's creators) thinks my one is the right size and big enough," explains Mike. His confident stance is not easily merited:



Quick **FACTS**

- The andotrope's display looks like a normal screen
- It's omnidirectional and has no backing or sides
- Which is ideal for groups viewing it at once
- The 60% scale replica weighs 13kg and is 71cm tall...
- ...and now sits in the HQ of games company Cyan, which made *Riven*

> At 1,200rpm, persistence of vision kicks in, creating an illusion of movement just like video



▲ Construction was a lengthy process

▶ A still from *Riven* showing the Gehn's Imager Mike has replicated

Mike is a self-taught maker and IT specialist who drives a DeLorean to maker events where his take on *Myst*'s linking book has gone down a storm: **magpi.cc/mystbook**.

Mike had not made anything with Raspberry Pi before, but it seemed ideal for the andotrope because he "needed a small, powerful computer with a lot of capabilities. Something that can blink lights on and off very well wouldn't cut it for this project. An Arduino or ESP32 microcontroller just isn't powerful enough, and a full desktop x86 computer would be such a hassle". Raspberry Pi is used as a central computer to act as the master control device – the main hub that controls and synchronises everything. It handles the user input and the local Wi-Fi network, it communicates to the tablets via MQTT, it manages the Arduino Pro Mini motor controller and DFRobot DRI0042 motor driver, and it plays back the audio. "I'm quite impressed with how much it can manage simultaneously!" says Mike.

Mike used ChatGPT as the starting point for his code, manually editing it, "fixing the edge cases" and expanding its scope. He then focused on finessing the displays so the two tablets were synchronised and phone photos and videos looked good. There were also tweaks to the andotrope's rotational speed and adjustments to the width of the viewing slit to improve the illusion of movement.

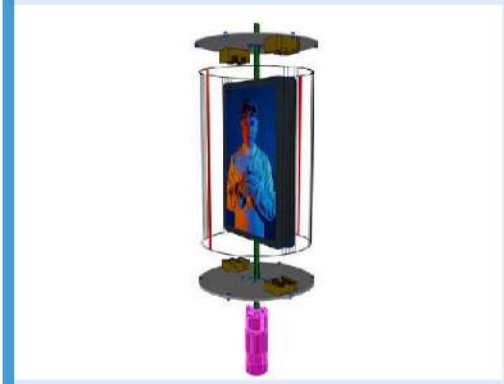
Brassed off

"Brass is a beautiful metal but I find it a pain to work with," says Mike. Designing and creating a means to securely hold all the brass rods in place was "a major headache", and the andotrope mechanism was also a challenge. "It's not that hard to get one of those displays working, but it's a lot of effort to get it working well."

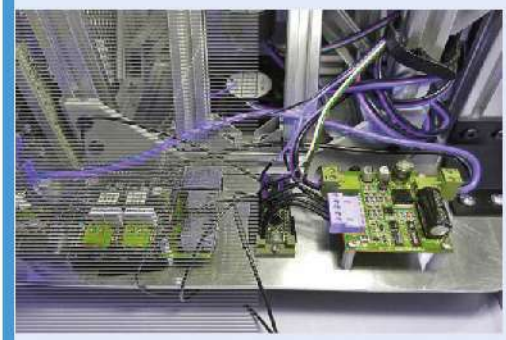
The andotrope project ended up "pretty expensive thanks to all that brass!" Mike notes that the brass sphere alone was around $700. The 10in Android tablets he chose were also pricey, but use very little power when sitting idle and "happily last a full nine hours of constant playing at a convention with battery life to spare," which he finds pretty amazing.

There are a lot of different uses for the Andotrope display – everything from a personal digital assistant (Siri/Alexa) display to a teleconferencing system to playing board games with friends. Anyone trying to create their own should use a phone or a tablet with zero flicker to its video, as any flicker at all will show up at high rotational speeds.

Use your illusion



01 The back-to-back Android tablets, motor and stepper controller, plus wireless access are all Raspberry Pi 3B-controlled.



02 Toggles on the outside of the case trigger the content to play, while Raspberry Pi tells the Arduino motor controller to spin the screens at a rate that ensures only one animation (length unimportant) is shown at once.

03 Mike designed a 3D printed case and spent hours fitting all the components and brass struts in place. A Python script runs on startup to control everything, with the two screens rotating at 1200rpm to simulate movement and be viewable from any angle.

Twirly Shirley

This remote-controlled precision turntable has **Phil King** spinning around

MAKER

Martin Spendiff and Vanessa Bradley

VEEB began as a lockdown photography project for Vanessa Bradley and Martin Spendiff, but soon morphed into a site documenting their builds.

veeb.ch

Warning!
Moving parts

Be careful when handling this project because it has moving parts. Children should be supervised.

Precision turntables are ideal for rotating items to photograph, among other uses, but can cost hundreds of pounds. So VEEB Projects' $50 DIY precision turntable – using Raspberry Pico W, 3D printed parts and a stepper motor – is all the more remarkable. It's actually an upgrade from the duo's previous turntable that featured an IR remote.

"We already made a turntable about a year ago to solve a problem we were having taking photos of something," reveals Martin Spendiff. "We wanted a turntable, but one we could control programmatically, so that we could be flexible in generating stop-motion footage."

That first turntable used a DC motor, which lacks precision, resulting in inconsistent movements which vary with the weight of the object placed on it. "We wanted something more precise," recalls Martin, "and we wanted a remote that we didn't keep losing, so we put it as a page on a web server."

In addition, they simplified the original version's 3D-printed gearing system, which included a worm gear, to use two gears instead of three.

Web-based spinning

Raspberry Pico W has a web server running on it, based on Simon Prickett's Phewap project (**magpi.cc/phewap**). On that server sits a page that shows a virtual remote. "The buttons trigger some MicroPython code that tells the turntable's stepper motor to turn, then 3D printed gears translate the movement from the motor to the bearing, which in turn moves the turntable." The 'lazy Susan'-style bearing, as used in turntables for the dining table, features multiple metal balls between two circular metal plates for low-friction movement.

▶ With the outer gear and bearing removed, you can see the stepper motor and Raspberry Pi Pico W

▲ By taking photos at set intervals, you can create stop-motion videos of a rotating object, such as a roller-skate

> "It was one of those things that you keep going back to when you learn something new during another project"

Since the stepper motor they used requires a fairly hefty voltage, a 20V Power Delivery trigger board is used to step up the voltage supplied via USB-C. "Using PD triggers was one of the good bits: it makes for a much smaller build than having to try to cram a power brick in there too. They're really cheap too, so it's a rare win-win."

VEEB chose Raspberry Pi Pico W for the project due to "a flat refusal to use C, due to lack of skill, and the knowledge that the documentation for Raspberry Pi stuff is less sparse than for other microcontrollers," says Martin. "When we get stuck, there's usually someone that has run into a similar problem and documented it somewhere."

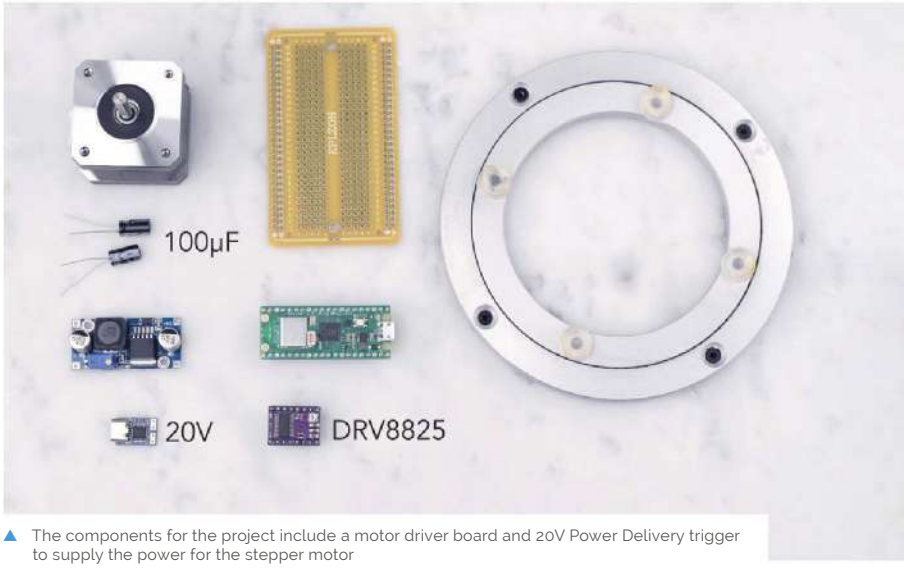
Quick FACTS

- The Twirly Shirley name is a play on 'lazy Susan'
- The stepper motor divides a full rotation into 6400 steps
- It's driven by a DRV8825 breakout board
- VEEB may consider adding physical control buttons and a display
- They'd also like a more elaborate web page remote

Learning process

Martin reckons the project took around a year to complete, working on and off. "There were a few mini projects in there: learning enough OpenSCAD for the gears, learning enough CSS for the remote page, getting the stepper motor to work. It was one of those things that you keep going back to when you learn something new during another project."

An impressive stop-motion video of a roller-skate and disco ball for the Twirly Shirley YouTube video (**magpi.cc/twirlyshirleyyt**) was shot using a camera taking photos at set intervals, but a possible upgrade would be to use a Raspberry Pi single-board computer to automate the process. "An SBC could take photos and send requests for the turntable to move," notes Martin. "The last thing we made used web sockets to make a Pico listen for signals being broadcast from a [Raspberry] Pi with a camera attached."



▲ The components for the project include a motor driver board and 20V Power Delivery trigger to supply the power for the stepper motor

Video magnifier

Don't hunch over your books again thanks to this marvellous, camera-powered magnifier using Raspberry Pi. **Rob Zwetsloot** zooms in

MAKER

Markus Gräser

A German software developer who has recently returned to Berlin after a two-year stint in London.

magpi.cc/magni

We know plenty of folks who love to whack up the font size on their ebook reader of choice, and we've definitely on occasion tried to pinch zoom printed media (do not judge us), so the idea of being able to zoom in on a book without hauling out a magnifying glass is very attractive. This is what maker Markus Gräser has achieved.

"[It's a] technical version of a magnifying glass to help people with low vision," Markus tells us. "It's basically a camera that can be connected to any HDMI screen, with a simple interface to scale and modify images. There are lots of professional devices out there, and a few DIY takes on the category as well. My goal was to make it simple, portable and affordable. There's also experimental support for reading out text."

▲ The case has easy access to power and AV output, and it even has USB access if you need more input options

Markus came up with the idea when his grandmother was prescribed one of these professional devices after suffering from macular degeneration.

"Even though she generally shunned away from most technology beyond a telephone or TV, this improved her life a lot and she enjoyed reading and writing again, be it letters, books or the newspaper," Markus says. "However, it's a bulky device that was placed in the living room, so when she needed it for cooking recipes she had to frequently go back and forth between the kitchen and the living room."

According to him, a DIY solution is at least a tenth of the price of professional versions.

Rapid prototyping

While a tablet or laptop would have done the job, the constant updates and unlock codes felt like a barrier. Raspberry Pi with a Camera Module and a screen seemed like the best solution to Markus.

"The software was relatively easy," Markus explains. "But I experimented a lot with the form factor. I started out with a wooden setup as I wanted to make it look nice and non-technical,

▲ Testing out the concept using a wooden board and a bottle of soda

but my latest version is 3D-printed and I'm quite proud of its simple and portable design."

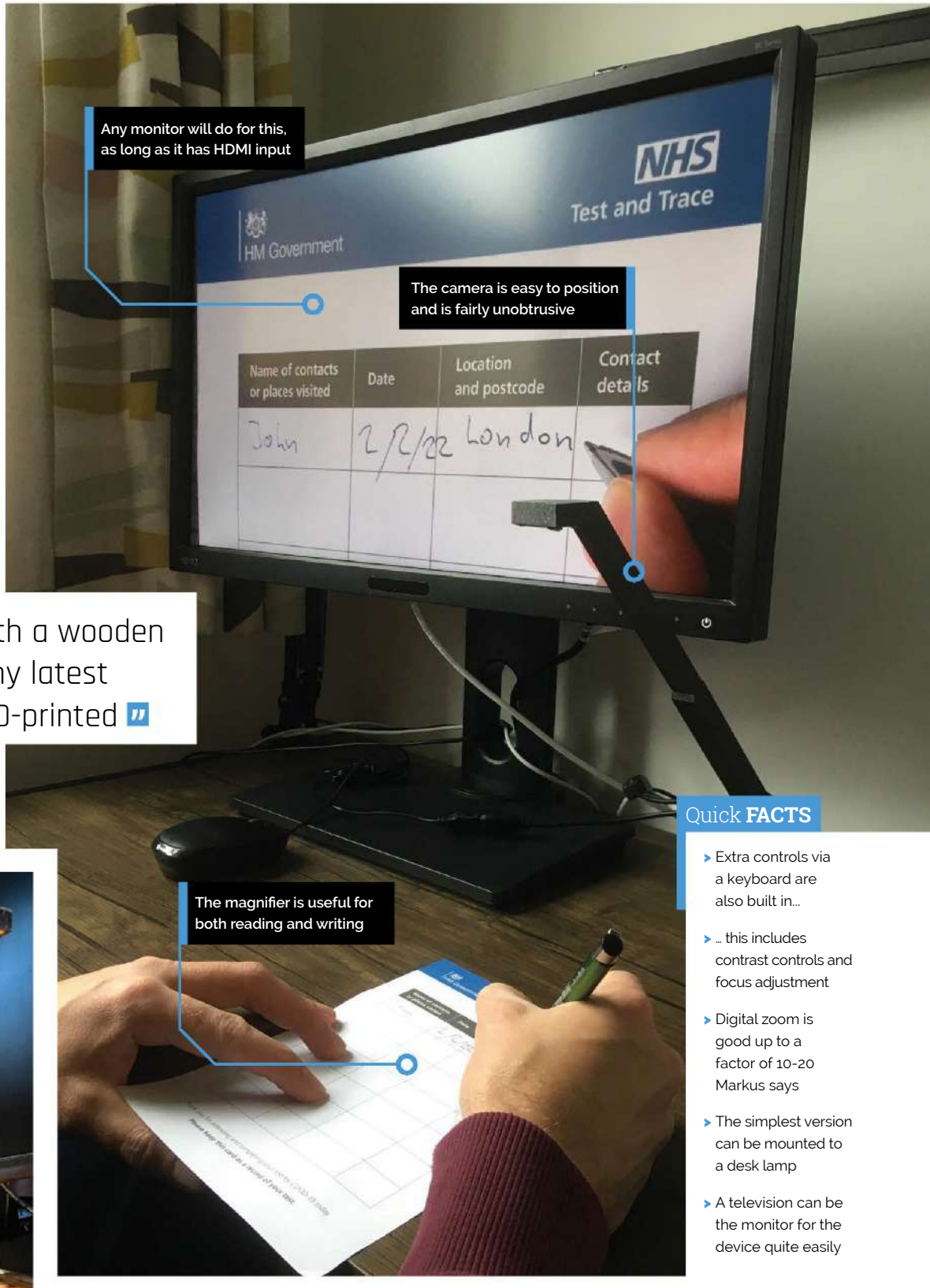
There are three main functions for the magnifier: a button cycles through different magnification levels, while another can change colour modes. This includes inverting the colours or tinting it yellow or blue.

"There's also an experimental readout feature," says Markus. "[It takes] a picture and then uses Tesseract OCR to get the text which is later read out via the pico TTS library, all on-device without the need for an internet connection."

Book smarts

"This has been an interesting project where I've learned a lot over the years," Markus mentions. Everytime he revisits it he thinks about little tweaks he can make but is happy that the code is up on his GitHub (**magpi.cc/magni**) so others can modify it to their needs.

Unfortunately Markus' grandmother had a stroke before getting to test it, and is now unable to use it. However, other folks have been able to give feedback, much of it positive. He's gone on to use Raspberry Pi and Camera Modules for other projects, such as a portable photo booth for weddings – sounds like something we need to cover in a future issue.



> I started with a wooden setup but my latest version is 3D-printed

Markus went through several 3D-printed designs for the device

Quick **FACTS**

- Extra controls via a keyboard are also built in...
- ... this includes contrast controls and focus adjustment
- Digital zoom is good up to a factor of 10-20 Markus says
- The simplest version can be mounted to a desk lamp
- A television can be the monitor for the device quite easily

SUCCESS STORY magpi.cc/success

Revolution Pi industrial computers

Powerful, flexible computing for industrial IoT and automation.

By **Rosie Hattersley**

Kunbus GMbH spotted a need for devices that marry reliable performance fit for industry with the flexibility and ease of consumer computing. They matched Raspberry Pi with custom software, enclosures, expansions, and peripherals to stunning effect.

Kunbus began in 2008 as a company that made discrete communications components for industry. Business Developer and founder Boris Crismancich characterises the pre-Internet-of-Things era as one in which there was a distinct separation between PCs — personal computers used by consumers and end users in office settings — and hardware used by industrial makers, who saw the unreliability of largely Windows-based devices and steered well clear.

▼ The RevPi Connect 4 expansion module

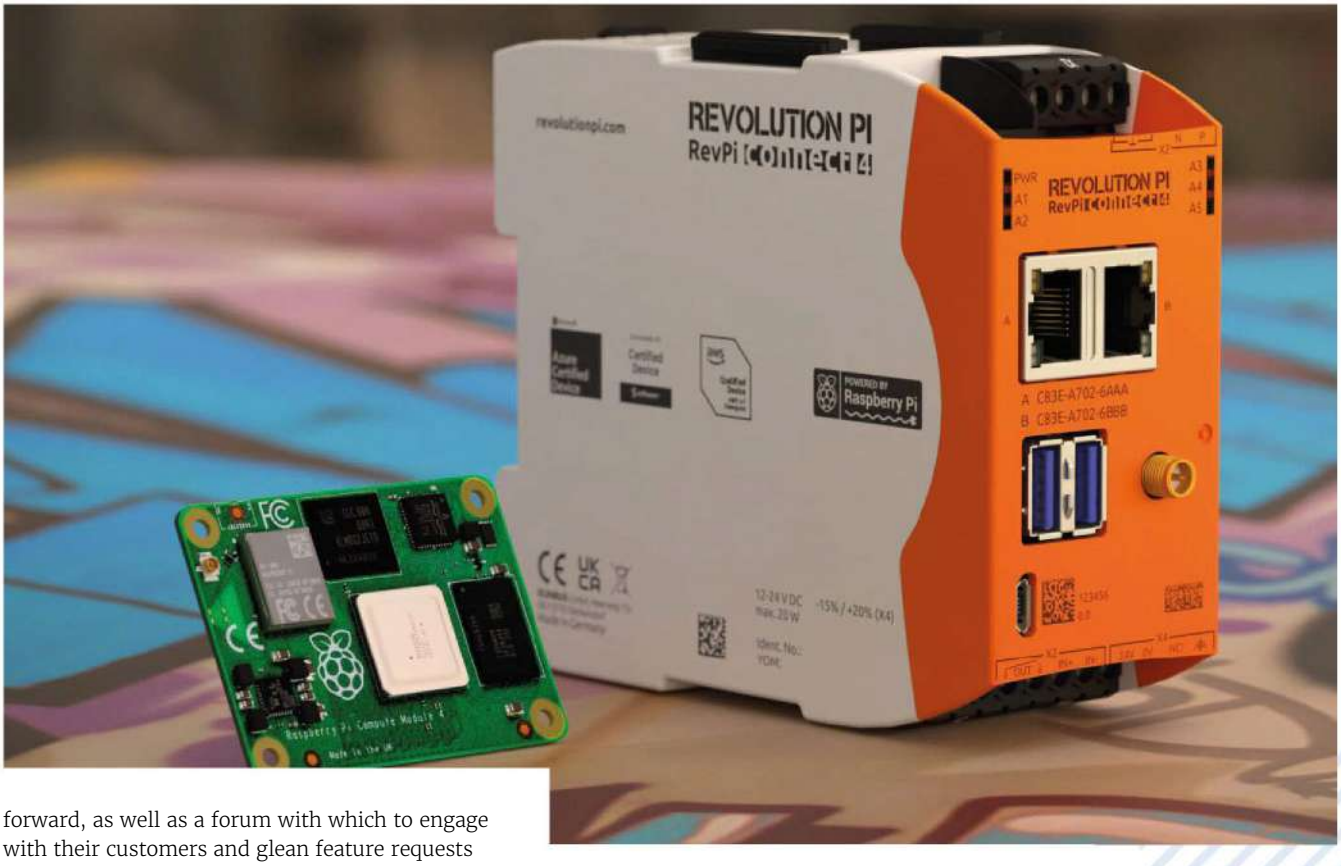
Industrial controllers, on the other hand, were prized for their reliability but avoided the name tag of computers. "There was a culture clash. The Internet of Things made PCs and controllers need to get along and Raspberry Pi was common ground."

Kunbus soon realised that alongside the need for industrial networks, there was demand for an industrial version of Raspberry Pi. By this time – early 2016 – Raspberry Pi had sold nearly seven million units, and was the darling of single board computing. The product was known and embraced by software engineers and developers, and it had the huge advantage of a strong community of enthusiasts keen to support others' development endeavours.

Kunbus saw the potential for an industrial Raspberry Pi controller model that offered the more robust specifications that OEMs needed, including an open source modular design, support for 24V power supply, DIN rail mounting, I/O expansion modules, and optional Ethernet, alongside a custom operating system. Raspberry Pi had not yet released its own industrial Raspberry Pi Compute Module at this stage, but the concept made sense and, with an established product line of communications devices, the team at Kunbus decided to forge ahead with designing what was to become Revolution Pi (**revolutionpi.com**) and a brand-new product line.

The challenge

Conversations at trade fairs led Crismancich to realise there was a real appetite for an open source device that was agnostic of any particular industry sector, and could instead be used almost universally and in ways that only the purchaser might have envisaged. In this respect, the device Kunbus imagined had much in common with Raspberry Pi itself. Rather than a challenge to overcome, Kunbus saw developing an industrial Raspberry Pi product as an opportunity. They were well placed to do so and had the experience and skills to take the project



forward, as well as a forum with which to engage with their customers and glean feature requests and feedback.

Head of marketing Ekkehard Krebs described the unheard-of ten-month development process, in 2016, of what became Revolution Pi – the first industrial controller based on Raspberry Pi. His colleague Nicolai Buchwitz, himself a former Revolution Pi customer in his previous role at a motorway freight weighing station, regards it as a direct response to the call for an industrial Raspberry Pi.

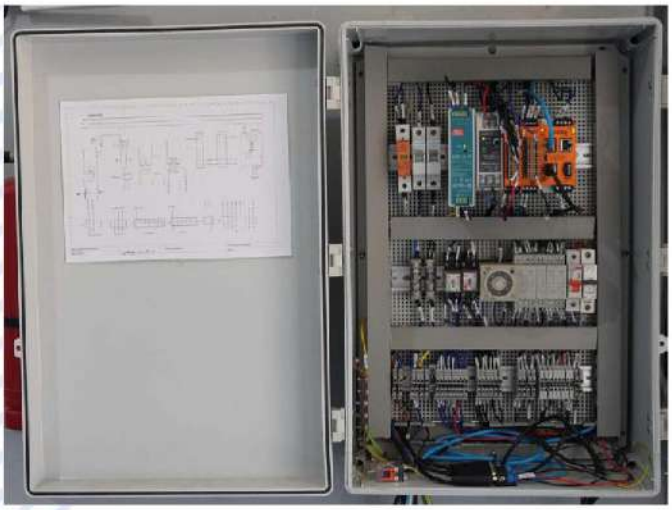
Why Raspberry Pi?

That Raspberry Pi would be the basis of Revolution Pi was never in question: there was no other device that had the market penetration, availability, and uptake within the open source community. Alongside this was an ever-expanding number of engineers and developers who, either for their own interests or through their studies, had used and created projects based on Raspberry Pi. This provided a familiarity with the base product, and furnished a potential market for Kunbus' new business model.

Crismancich says that other controller companies were still totally focused on closed, proprietary devices, much as consumer brands were at this time. Although Raspberry Pi is not fully open hardware, the company's commitment to sharing code and schematics wherever possible, together with the adaptability of the platform — pretty much anything open source was compatible with Raspberry Pi by design — made it unique. Different architecture would not have worked, believes Crismancich, because the IT world didn't know about other hardware and was accustomed to operating with user input, whereas the IP (industrial process) world was all about automation. "They needed to find common ground."

Two months after Kunbus launched Revolution Pi, Raspberry Pi announced its first Compute Module – its own industrial Raspberry Pi – and Kunbus knew they were on to something. They quickly switched over to using Compute Module 1, then a single-core computer running at 500MHz, recalls Ekkehard Krebs, and have been a major customer of CM1 ever since. Since CM3 had the same form factor and pinout (detailed on the Raspberry Pi website, of course), upgrading to this model and adding a Kunbus carrier board and custom OS was straightforward. After a change in form factor between CM3 and CM4, the upcoming CM5 again retains the CM4 form factor, and large industrial customers like Kunbus have been involved and kept abreast of the development process so they can launch their own new products, extension boards, and software concurrently.

▲ RevPi Connect 4 sitting alongside Raspberry Pi Compute Module 4



The solution

The three main Revolution Pi options are powered by Raspberry Pi Compute Module 4, with options for 2GB, 4GB or 8GB RAM and either 8GB or 32GB SD storage, together with two Gigabit Ethernet ports and USB 3.2, micro USB, and micro HDMI ports. A PiBridge allows for the addition of two I/O and comms-focused expansion boards as well as fieldbus gateways. Each RevPi comes in a slim DIN-rail housing, with the option to customise this enclosure with logos and other branding.

> Kunbus saw the potential for an industrial Raspberry Pi controller model that offered the more robust specifications that OEMs needed

The hardware is certified for use with all major cloud services, and engineered to work seamlessly with sensors and securely process their data. All models have Kunbus' custom version of Raspberry Pi OS plus the popular IoT tool Node-RED, a click-to-assemble tool known as PiCtory that supports communication between RevPi and connected devices, and the RevPiModIO Python library.

Kunbus Product Owner Nicolai Buchwitz calls RevPi "the Swiss Army knife of automation and IoT". He explains that because of the retained similarity with the Raspberry Pi hardware on which it is based, there is an almost endless number of software components that can be used with RevPi. Customers can choose or assemble applications from many packages. Fieldbus gateways connect with other industrial components in a network and allow them to communicate.

▲ An industrial control panel using Revolution Pi hardware

The results

Creating an industrial product line around Raspberry Pi couldn't have gone much better for Kunbus. The company has sold somewhere between 150,000 and 160,000 units and seen its Revolution Pi used in everything from solar power units and electric race cars to hydrogen filling stations, and in settings from Nintendo to micro breweries. Crismancich learned of the latter at the Hanover industrial trade show when the owner saw Kunbus on his name tag and showed him what was at the heart of the beer rig. In fact, it turned out that nearly half of the exhibitors had RevPi in their products.

Cloud certification for the likes of Microsoft Azure and Amazon Web Services by Kunbus' engineers for their customers works like a dream, says Krebs, because of the ready-to-use examples that work with Raspberry Pi. Kunbus now has ten such engineers, some of whom are actively involved in developing the Linux kernel on which Raspberry Pi OS and Revolution Pi's custom OS are based, and 150 employees. Krebs notes that the young engineers who are now coming through learned about Raspberry Pi at school, used it during their studies, and are confident choosing it as the basis of their own companies' products.

Crismancich also points to the millions of software engineers and developers using Raspberry Pi worldwide and the 50 million plus Pis sold.

The might of Raspberry Pi as an organisation, its rock-solid community and support network, and the guarantee that the products customers use now will still be available and supported a decade from now all build trust.

Kunbus was "almost first, if not first, to market" with an industrial Raspberry Pi, and of course has imitators. Rather than considering them rivals, having such a solid customer base means Kunbus is able to regard these other hardware companies as 'marktbegleidet' – companions or fellow travellers on a mission to make everything open source. The market is large enough, and Kunbus can afford to be generous.

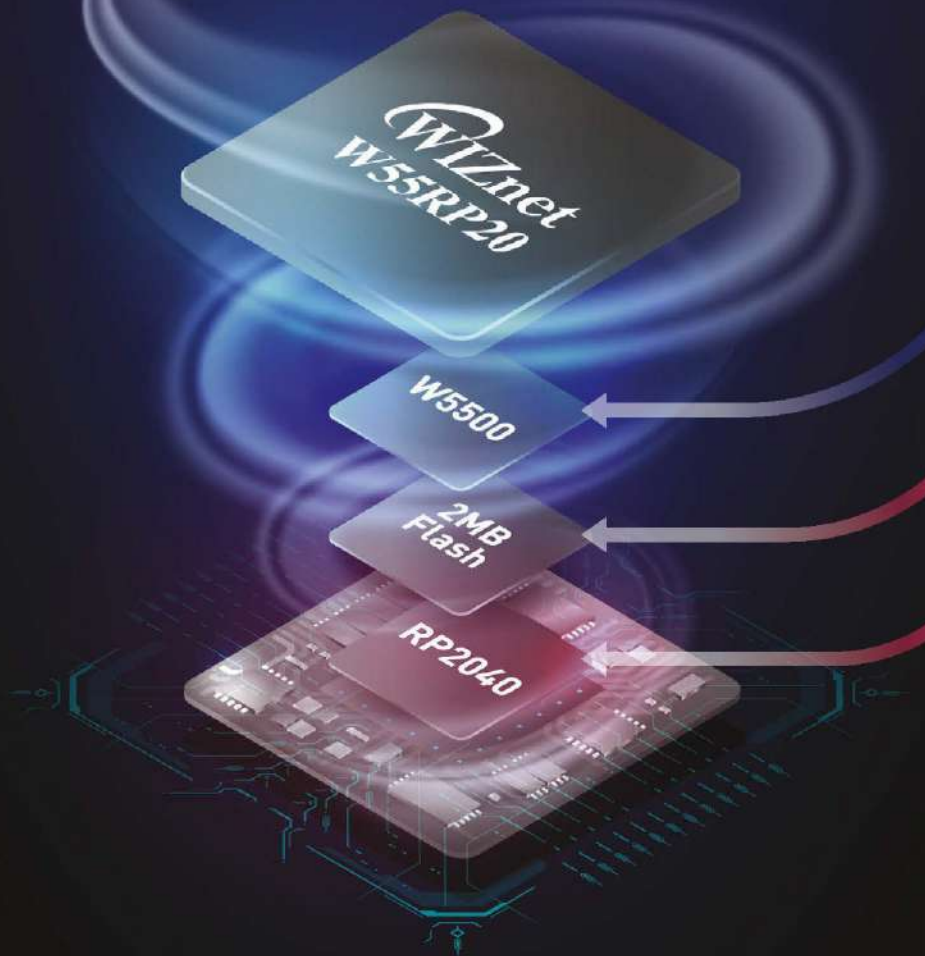
▶ Inside the RevPi Connect 4 you'll find a Raspberry Pi Compute Module 4

WIZnet

IoT Network Accelerator

ioNIC

Co-Processor



internet offload NIC

Features	Advantages
All RP2040 and W5500 Features	Designed for small form factor devices
Integrated TCP/IP Stack, MAC & PHY	Minimized circuit design complexity
2Mb Ultra-Low power flash	Lower assembly cost & time to market

TOE(TCP/IP Offload Engine)

Ethernet MAC/PHY

Pre-Programmed & Custom Service

Modbus	Serial	CoAP
TLS	WebServer	Cloud
HTTP(S)	MQTT(S)	OTA

Raspberry Pi
Design Partner

Meet the New Evaluation Board

COMING SOON **$9.95**

For reselling & bulk purchases, please contact direct@wiznet.io

SUBSCRIBE TODAY FOR JUST £10

Get 3 issues + FREE Pico W

+

Subscriber benefits

- **FREE Delivery**
  Get it fast and for FREE
- **Exclusive Offers**
  Great gifts, offers, and discounts
- **Great Savings**
  Save up to 23% compared to stores

Subscribe for £10

- Free Pico W
- 3 issues of The MagPi
- £10 (UK only)

Subscribe for 6 Months

- Free Pico W
- 6 issues of The MagPi
- £30 (UK) $43 (USA)
  €43 (EU) £45 (Rest of World)

Subscribe by phone: **01293 312193**

Subscribe online: **magpi.cc/subscribe**

Email: **magpi@subscriptionhelpline.co.uk**

Subscribe for £10 is a UK-only offer. The subscription will renew at £15 every three months unless cancelled. A free Pico W is included with a six-month subscription in USA, Europe and Rest of World.

ESSENTIAL ELECTRONICS

Build your first simple circuits with a handful of cheap components

Sean McManus

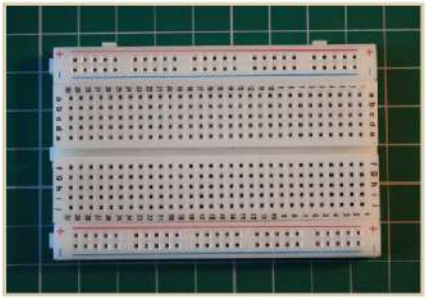
The magic of Raspberry Pi is that it can interact with the world. You can gather input from buttons and sensors that detect light, movement or distance. Send output through lights, buzzers or motors that drive robots around the room.

It's easy to get started. Using a breadboard, you press components and wires into holes to build a circuit, with no need for soldering. Jumper wires make it easy to connect your circuit to Raspberry Pi's general-purpose input/output (GPIO) pins.

The GPIO Zero library makes it easy to accept input and control your circuits from Python.

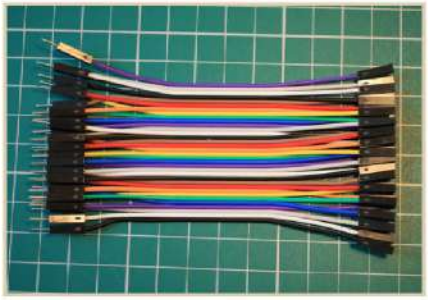
SHOPPING LIST

Discover basic components you can combine to make a variety of projects



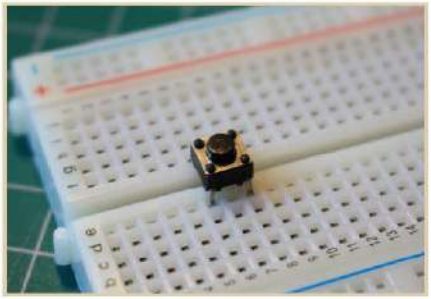
Breadboard

Ideal for prototypes and experiments, breadboards enable you to connect components without soldering. Simply press them into the holes on the board.



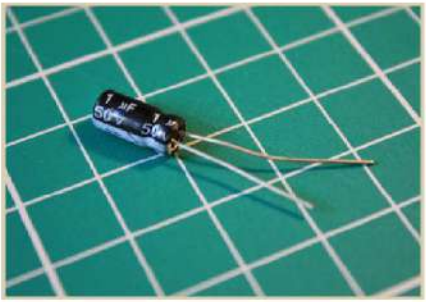
Jumper wires

Use socket-to-pin jumper wires to connect a Raspberry Pi GPIO pin to the breadboard. Pin-to-pin wires connect between points on the breadboard.



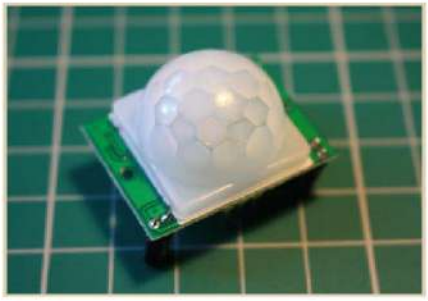
Button

Pressing a button can complete a circuit to, for example, turn a light on. Buttons are great input devices for Raspberry Pi and a vital part of your collection.



Capacitor

Capacitors can be charged up to store small amounts of electricity, which they can later discharge into the circuit. They're often used to make timers.



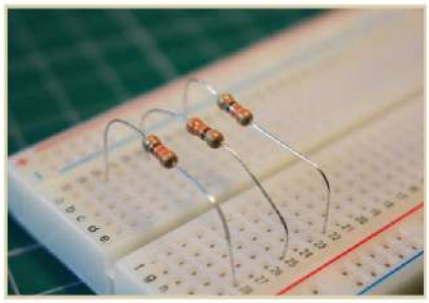
Passive infrared (PIR) sensor

Motion sensors like this are often used in burglar alarms or automatic lights. These are great for triggering spooky Halloween projects when someone approaches.

Ultrasonic distance sensor

Essential for robots detecting walls or obstacles, this distance sensor by Kitronik measures distances of between 2cm and 400cm, with accuracy as good as 3mm.

> Passive buzzers give you more flexible pitch control, so you can program tunes

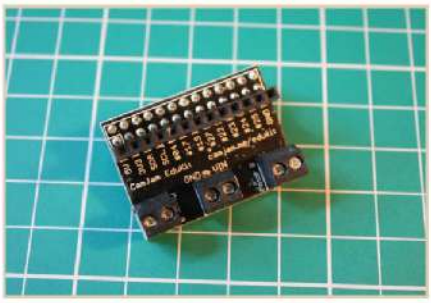


Resistors

Resistors reduce current flow in a circuit. Coloured stripes indicate the resistor's value. Find decoders online (see **magpi.cc/resistorcalc**) or see the table in the *Raspberry Pi Beginner's Guide*.

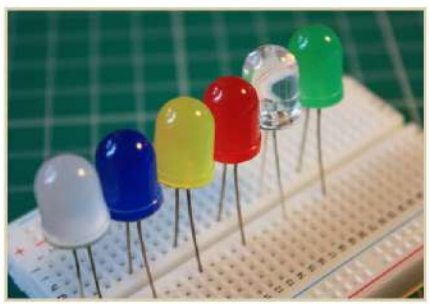
Buzzer

Active buzzers are easiest to wire up but usually just produce one tone. Passive buzzers (also known as tonal buzzers) give you more flexible pitch control, so you can program tunes.



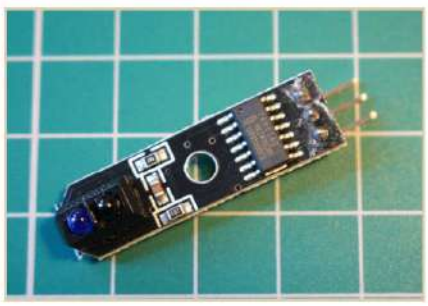
Motor board

A motor controller board, such as this one from the CamJam Robotics EduKit, sits between your Raspberry Pi's GPIO pins and your project's motors. It allows you to control things like robot wheels.



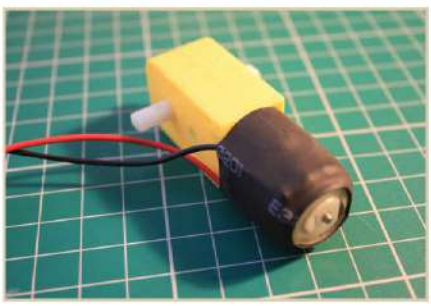
Light emitting diodes (LEDs)

Lights! They come in a range of colours and sizes, and they add sparkle and joy to any project. Place them carefully: they only work one way around.



Line-following sensor

This sensor has both a light emitter and a light detector. Using it, you can create a robot that can follow a black line drawn on a white surface.



Motor

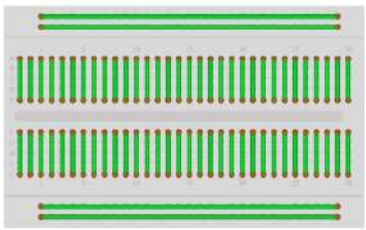
Essential for mobile robots, DC motors like this connect to your Raspberry Pi through a motor controller board. To save time, get one with pre-soldered wires.

CREATE A MORSE CODE TRANSMITTER

Wire up an LED to Raspberry Pi, and flash the bulb to send messages

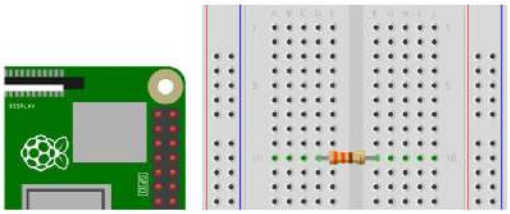
01 Understanding the breadboard

Inside the breadboard are electrical connections between the holes, as shown in this picture. In the middle of the breadboard, there are numbered rows. The holes in a row that form a group are connected to each other. Groups are usually separated by a trench.



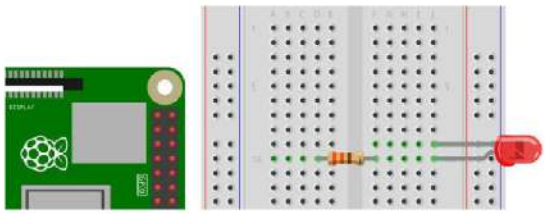
02 Connecting the resistor

In our circuit, current flows from Raspberry Pi's GPIO pin, through a resistor and LED, and back to Raspberry Pi's ground pin. We use a 330 ohm resistor to stop the LED drawing too much current, which might damage the LED or Raspberry Pi. To force the current to go through the resistor, we have to connect each end of it to a different group of breadboard holes. A resistor can go either way around.



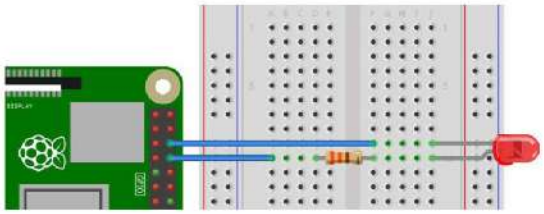
03 Connecting the LED

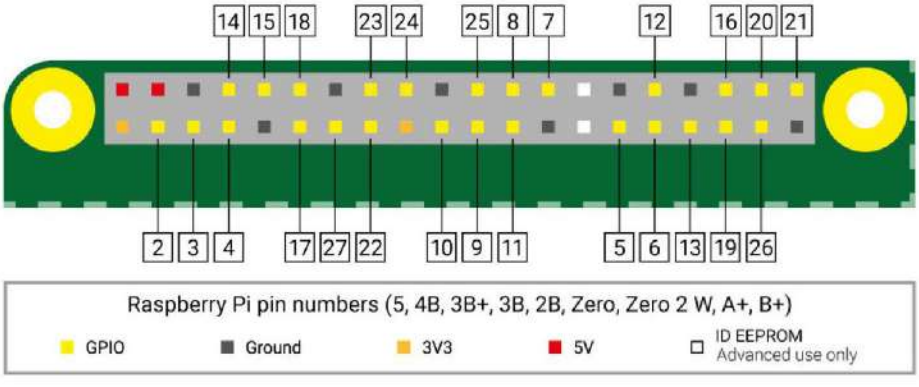
The LED's long leg is the anode (positive end), and its short leg is the cathode (negative end). Connect the anode to the same group of breadboard holes as one end of the resistor. Connect the cathode to a new group of holes.



04 Connecting Raspberry Pi

Power off your Raspberry Pi before connecting wires or components. The GPIO header has pins that provide 3.3V or 5V of power constantly, general-purpose input/output (GPIO) pins, and ground pins for completing a circuit. To control our circuit, we'll use GPIO pin 14, which is the fourth pin on the top row. Connect a jumper wire between that pin and the resistor, at the opposite end to the LED. Connect a ground pin to the LED's negative leg.





◀ We use BCM pin numbers, which aren't the same as the physical pin locations

MORSE CODE TRANSMITTER PROGRAM

For if you always wanted to learn Morse code, but never did it. Da-dit, da-dit, etc...

240625 magpi morse code.py

DOWNLOAD THE FULL CODE:
magpi.cc/seancode

Language: **Python**

```
001.  from gpiozero import LED
002.  from time import sleep
003.
004.  led1 = LED(14)
005.
006.  def blink(flash_duration):
007.      led1.on()
008.      sleep(flash_duration)
009.      led1.off()
010.      sleep(0.5)
011.
012.  def flash_message(message):
013.      for symbol in message:
014.          if symbol == ".":
015.              blink(0.5)
016.          if symbol == "-":
017.              blink(1.5)
018.          if symbol == " ":
019.              sleep(1.5)
020.
021.  flash_message(".... . .-.. .-.. ---")
```

> Send our Morse message to the flash_message function

Line 1: Set up GPIO Zero so we can use the LED code

Line 2: Import the time library to control the length of light flashes

Line 4: Set up led1 as an LED on GPIO pin 14

Line 6: The blink function receives the flash duration in seconds

Line 7: Turn the LED on!

Line 8: Pause for as long as we want the light lit

Line 9: Turn the LED off

Line 10: A short pause stops the flashes merging

Line 12: This function receives a Morse code message

Line 13: The loop looks at each character in the message

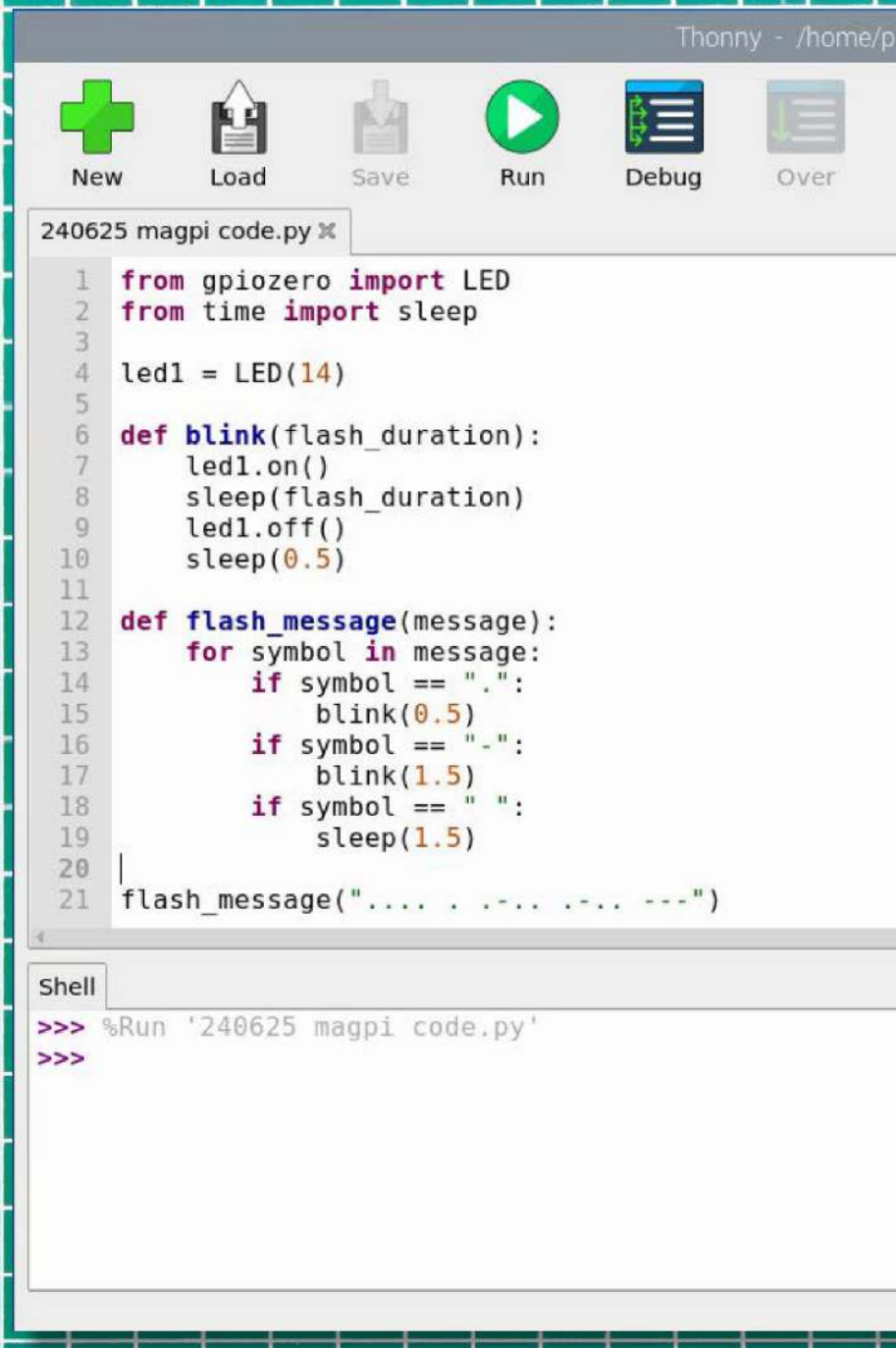
Line 14 and 15: If the character is a dot, we blink for half a second

Line 16 and 17: If it's a dash, we blink for 1.5 seconds

Line 18 and 19: If it's a space, we add a 1.5 second pause

Line 21: Send our Morse message to the flash_ message function

Run this program in Thonny by clicking the Run button. Get the code at **magpi.cc/seancode**.



▲ Find Thonny in the Programming folder of the Raspberry Pi OS menu

ELECTRONICS PROJECTS TO TRY

Discover how to use buttons, sensors and motors with these starter projects

GPIO Music Box

Components: breadboard, buttons, pin-to-socket and pin-to-pin jumper wires, speakers
magpi.cc/gpiomusicbox

Buttons are the simplest way to control circuits and collect input. In this project, you see how to connect four of them to Raspberry Pi. They're used to trigger sound effects played with Python's Pygame library, which you could customise with your own recordings. You see how to connect all your buttons to the same GPIO ground pin, by using the ground rail along the long side of the breadboard. With this approach, you don't need as many jumper wires to Raspberry Pi, so your circuit is easier to build and understand.

Quick Reaction Game

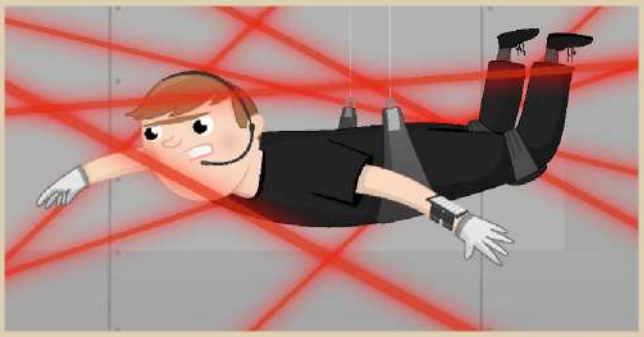
Components: breadboard, buttons, LED, resistor, jumper wires
magpi.cc/pythonqrgame

Who can push their button quickest when the light goes off? This two-player game uses a breadboard to connect two buttons and an LED to Raspberry Pi. With the music box, it doesn't matter in what order the buttons are pressed, but in this game you see how to detect which button was pressed first, and announce the winner on-screen. The program also shows how a single Python function can be used with GPIO Zero to provide different outputs for different buttons. You can extend the project by adding a timer or keeping track of scores.

Laser Tripwire

Components: breadboard, capacitor, light-dependent resistor (LDR), jumper wires
magpi.cc/lasertripwire

When intruders break the invisible beam, the alarm sounds. This project works by shining a laser pointer at a photoresistor, and sensing when the light changes because somebody has walked between them. GPIO Zero enables you to wait until light or dark is detected, and to run a particular function when it is. You can raise the alarm by playing a sound file through speakers, or by connecting a buzzer to Raspberry Pi.



Parent Detector

Components: PIR motion sensor, Raspberry Pi Camera Module, jumper wires
magpi.cc/parentdetector

The parent detector project is a great way to experiment with a PIR motion sensor. The sensor has pins that you can connect directly to your GPIO pins using socket-to-socket jumper wires, with no need for a breadboard. This project uses the sensor to trigger video recording, and only stops when no motion is detected. The PIR sensor has dials you can use to adjust the sensitivity and detection time of the sensor. Perhaps you could turn down the sensitivity and make a game where players must sneak up on the Raspberry Pi undetected?

Ultrasonic Theremin

Components: breadboard, ultrasonic distance sensor, tonal buzzer, jumper wires
magpi.cc/theremin

The theremin, patented by Leon Theremin in 1928, is an instrument that you play by moving your hands near it. It's famous for creating the woo-ooo sound effects in sci-fi films. You can make something similar by connecting an ultrasonic distance sensor and a buzzer to Raspberry Pi. The sensor sends out sound pulses, and measures how long they take to bounce back, so it can measure the distance to the nearest object. GPIO Zero includes simple code to measure the distance in metres, and to make a tonal buzzer play musical notes using the MIDI standard.

Robot Buggy

Components: Motor controller board, motors, wheels, jumper wires, AA battery holder and batteries for the motors, USB battery pack for Raspberry Pi, ball caster
magpi.cc/robobuggy

This *is* the droid you are looking for! Build your first robot by connecting a motor controller board, two motors and wheels to Raspberry Pi. A ball caster keeps everything balanced, acting as a central third wheel. The motors are powered by AA batteries. Log in to Raspberry Pi remotely using VNC or SSH and use a USB battery pack for power, so there's no need for monitor or power cables and your robot can roam freely. Thanks to GPIO Zero, you can use simple functions to move your robot forwards, backwards, left and right. You can extend this project by adding a line-following sensor (**magpi.cc/linefollower**). For motor boards and other components, see CamJam EduKit 3 and Monk Makes Servo Kit.

Part 02

Productive Python: Fun with files

Automate tiresome tasks with simple Python programs. Discover how to search PDFs, download web pages, and save the output.

MAKER

Sean McManus

Author of Scratch Programming in Easy Steps, Mission Python, and Web Design in Easy Steps. Get free chapters at Sean's website.

sean.co.uk

You'll Need

- PDFs to search **magpi.cc/issues**
- Raspberry Pi OS with desktop **magpi.cc/imager**
- Libre Office **libreoffice.org**

Last month, we experimented with clocks to improve productivity. This month, we'll see how you can delegate some common tasks to Python. First, we'll create a program that searches through your archive of *The MagPi* PDFs. It works for any PDFs, in fact – we've used it for bank statements and ebooks. The output is saved to a text file, which you can print, browse or keyword search. Then, we'll see how you can download web content and save it as a formatted file for LibreOffice. Next time you find you're missing a tool for file management, perhaps you can use the ideas here to build your own.

01 Create a new folder

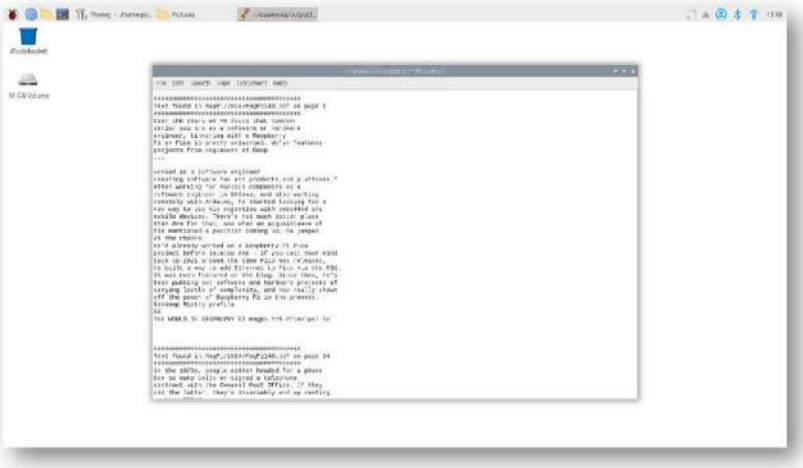
We'll need to install some Python libraries for these programs. Since the release of Raspberry Pi OS Bookworm (October 2023), you can't install libraries into the main version of Python. Instead, you create a virtual environment which contains a separate Python configuration. There, you can install all the libraries you want. Click the File Manager icon on the toolbar in Raspberry Pi desktop. When the File Manager opens, click the New Folder icon on its toolbar, give your folder a name, and click OK. This folder is for our virtual environment.

02 Create your virtual environment

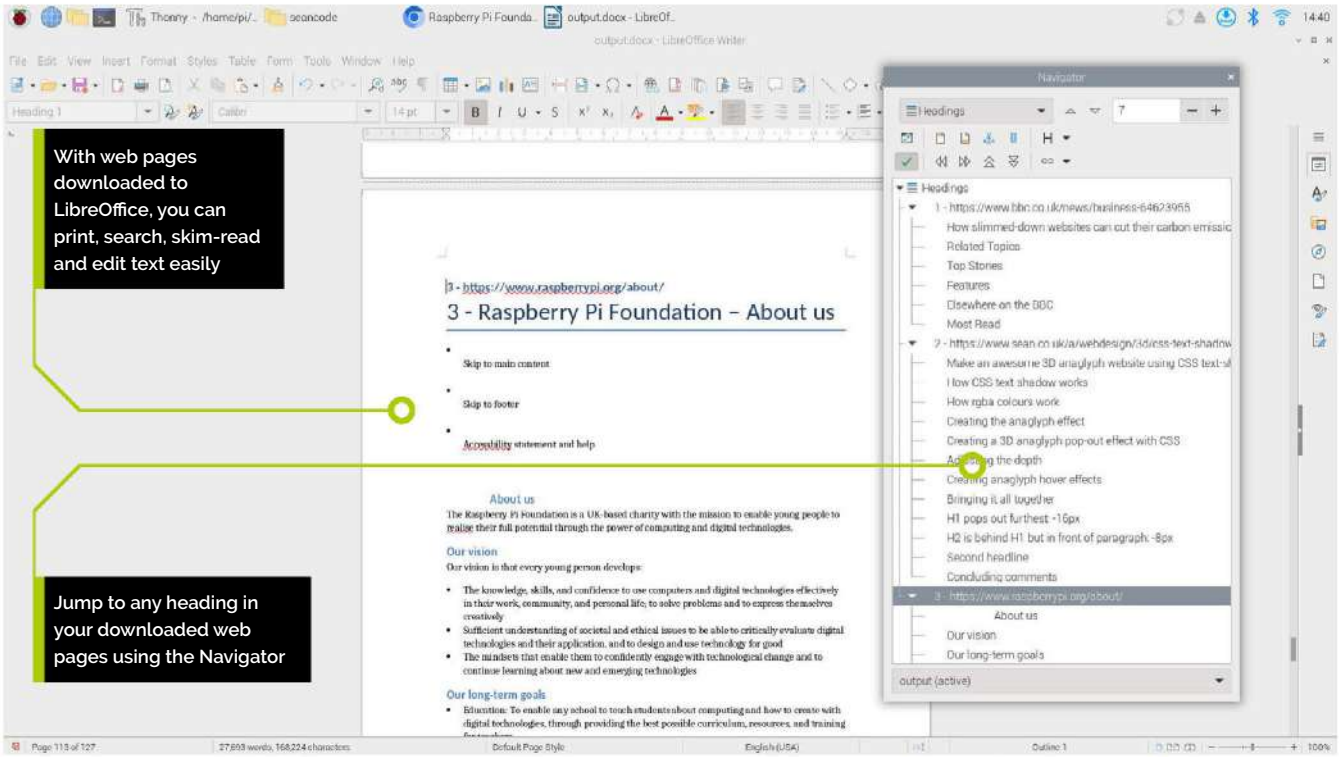
We'll use Thonny to build and run our Python programs. It comes with Raspberry Pi OS, and it has tools for installing and managing libraries (if you prefer to use the command line, see our tutorial in issue 136, **magpi.cc/136**). Open Thonny and click the three-line menu icon in the bottom right. Choose 'Configure interpreter'. When the dialog box opens, click 'New virtual environment'. Choose the empty folder you just created. The virtual environment is created and the 'Python executable' part of the dialog box is updated to show it's being used. Click OK.

03 Install the libraries

If there is no Tools menu, click 'Switch to regular mode' in the top right corner and restart Thonny. From the Tools menu, choose 'Manage



Your PDF search results are in a text file. The PDF excerpts show you the context for your search term

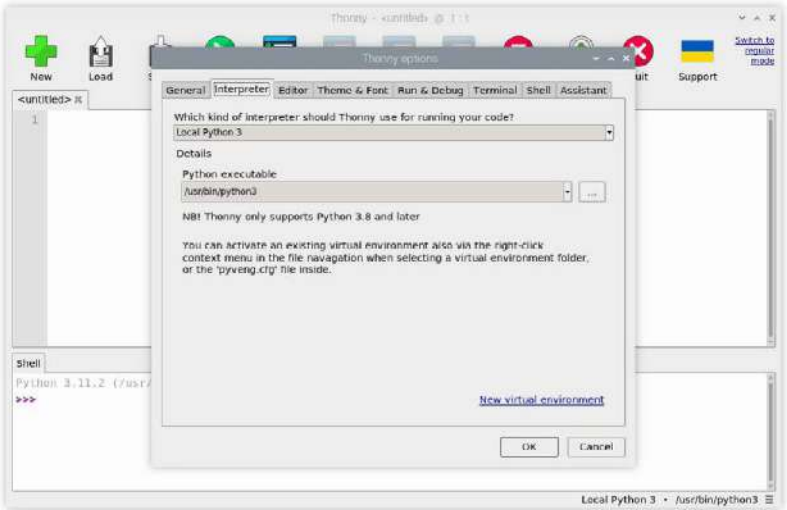


packages'. From here, you can search for Python libraries, install them, and uninstall them. It's much easier than using pip or apt on the command line. Use the search box to find 'requests', select it and then click the 'Install' button. Do the same for the bs4, python-docx and PyPDF2 libraries. Close the dialog box when you've finished.

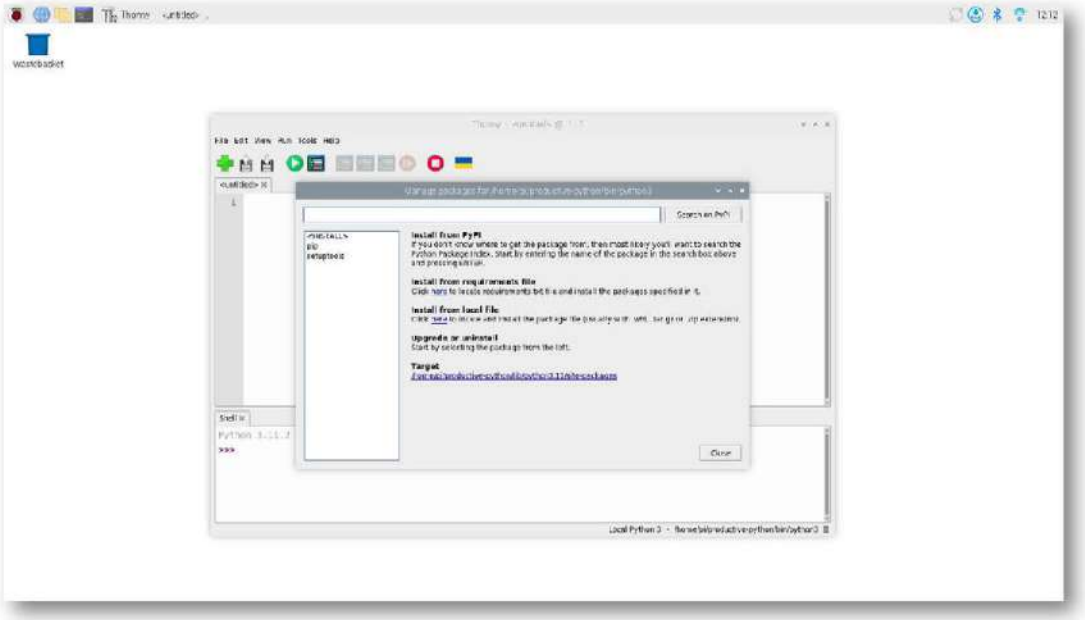
04 Get your files in order

Listing 1 searches through PDFs, to help you unearth past tutorials in *The MagPi*, or comb through ebook PDFs from sites like Humble Bundle. The program could search through every file on the Raspberry Pi, but it would be very slow. We'll speed it up by collecting together the PDFs we want to scour. Let's make a folder called **MagPi** and put our back issues there. Within that folder, you can organise PDFs into subfolders, for example by year. Your Python program and the **MagPi** folder should be side-by-side, for example in your **pi** folder if you have one. Don't put the listing inside the **MagPi** folder with the PDFs.

> The program could search through every file on the Raspberry Pi



▼ The Thonny options make it easy to create a new virtual environment so you can install Python libraries



Thonny's package manager makes it easy to install Python libraries with a visual interface

Top Tip

Speed up the search

Try only checking the first few pages of each PDF, where the table of contents is.

05 Searching through files

The Python os library uses the word 'directories' to refer to what the desktop calls 'folders'. The **process_directory()** function shows you how to spider through folders, looking for files of a particular type. We call it from line 34, giving it the name of the folder our PDFs are in. In the function, we use the os library to get a list of all the directories and files in the current folder/directory. For each one, if it's a directory, we call the **process_directory()** function again to look inside it. If it's a PDF, we call the **search_in_pdf()** function and give it the path to the PDF file. Other file types are ignored. The Shell pane is updated to show the program's progress.

06 Searching within the PDF

In the **search_in_pdf()** function, we open the PDF file and use the PyPDF2 library to process it. Lines 16 to 19 show how to search within each page of the PDF. When the program runs it asks you for the text you'd like to search for (line 31). Before checking whether the search term is on the PDF page, both the page text and the search key words are converted to lower case (line 19). This ensures we don't miss any matches because of inconsistent capitalisation.

07 Sending the output to a text file

Programs often have too much output to fit in the Shell pane, so we'll send ours to a text file. You can read it with the Text Editor, which is in the Accessories folder of the desktop menu. Lines 32 to 33 show you how to open a text file and write to it. Using with is the best way to open a file, because it automatically closes the file at the end of the indented lines under it. In line 34 we open the file in write mode (w), which starts a new file and overwrites any existing file of the same name. In line 20, we open the file in append mode (a), which adds onto the end of it.

08 Extracting text from the PDF

The output is sent to the file using **print()**. We're using f-strings (see line 23) to add the file path and page number from their variables. Using string slicing, we specify the start and end character from the **page_text** that we want to use. We output the first 200 characters (0:200) and then up to 450 characters before and after the search term (see lines 27 to 29). This excerpt gives you a good understanding of how the search term is used on the page.

09 Downloading web pages

We often need to compile online research into a single file so we can search it easily and read it offline. That's what Listing 2 does. It starts by using `stdin.readlines()` to ask for your list of links. Press **ENTER** after typing or pasting in each web page address. To finish, use **CTRL+D**.

10 Cooking up Beautiful Soup

Beautiful Soup is a library for parsing web files, which are coded in HTML. In HTML, text has short codes either side of it, called tags. These describe the start and end of paragraphs, headings, and so on. In lines 15 to 16, we download a web page and create a soup object for it. To help remove navigation links, lines 17 to 18 strip out any sections marked with nav and footer tags. Line 23 shows how you to find specific HTML tags in the web page. The program finds all the paragraph (p) tags, heading (h1 to h6) tags, table tags, blockquote tags and list bullet (li) tags.

11 Creating a docx file

The docx file format works in Microsoft Word and LibreOffice and can be created with the python-docx library. Line 10 defines our document object. Line 20 adds a main heading to it showing the web address. The 1 at the end of this line makes this a top-level heading. The page title which appears in the browser's tab is added in line 22. For each of the discovered tags we want, we check its name to see what type of tag it is. Headings are added to the docx file as headings, but they're all set to be a level 2 heading (line 25). That makes it easier to navigate the file in LibreOffice. List bullets are styled as bullets (line 27) and everything else is output as a normal paragraph. After each web page, we add a page break (line 30). Finally, line 31 saves the output.docx file, overwriting any previous version.

12 View with LibreOffice

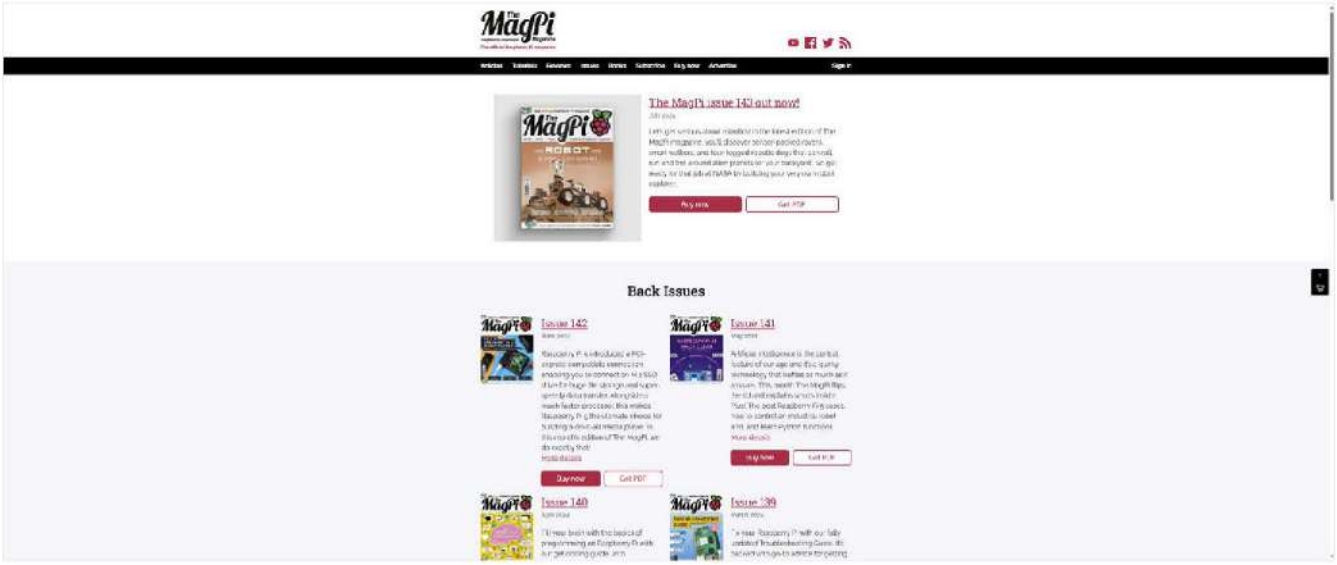
Open the Preferences folder in your desktop menu and choose Recommended Software. Install LibreOffice in the Office category. Double-click your output.docx file to open it in LibreOffice. Use **CTRL+F** to search in your document. **F5** opens the navigator. This enables you to jump between the documents and headings easily. Close the navigator again to edit the file. You can take the techniques in Listing 2 further to scrape information such as a share price from a web page and feed it to a Python program, or use python-docx for printer-friendly output from sensors.

Top Tip

Merging PDFs

PyPDF2 can be used to combine PDFs, too, including extracting specific pages.

▼ Download *MagPi* PDFs at **magpi.cc/issues**. When the search finds the articles you need, you can read them on screen or in your paper back issues



listing1.py

DOWNLOAD THE FULL CODE: magpi.cc/seancode

> Language: **Python**

```
# PDF search, with output sent to text file
import os, PyPDF2

def process_directory(path):
    for dir_or_file in os.listdir(path):
        path_plus_dir_or_file = os.path.join(path, dir_or_file)
        if os.path.isdir(path_plus_dir_or_file):
            print("\nProcessing subfolder:", dir_or_file)
            process_directory(path_plus_dir_or_file)
        elif dir_or_file.endswith('.pdf'):
            print("* Searching PDF:", dir_or_file)
            search_in_pdf(path_plus_dir_or_file)

def search_in_pdf(pdf_file):
    opened_file = open(pdf_file, 'rb')
    magazine_content = PyPDF2.PdfReader(opened_file)
    for page_number, magazine_page in enumerate(magazine_content.pages):
        page_text = magazine_page.extract_text()
        if search_string.lower() in page_text.lower():
            with open("output.txt", "a") as output_file:
                print("\n\n", file=output_file)
                print("#" * 40, file=output_file)
                print(f"Text found in {pdf_file} on page {page_number + 1}", file=output_file)
                print("#" * 40, file=output_file)
                print(page_text[0:200], "\n...\n", file=output_file)
                position_in_text = page_text.lower().rfind(search_string.lower())
                print(page_text[max(0, position_in_text - 450) :
                    min(position_in_text + 450, len(page_text))
                    ], file=output_file)

search_string = input(f"What term would you like to search for in the PDFs? ")
with open("output.txt", "w") as output_file:
    print(f"Ok! Searching for {search_string}", file=output_file)
process_directory("MagPi") # Change to your folder name
```

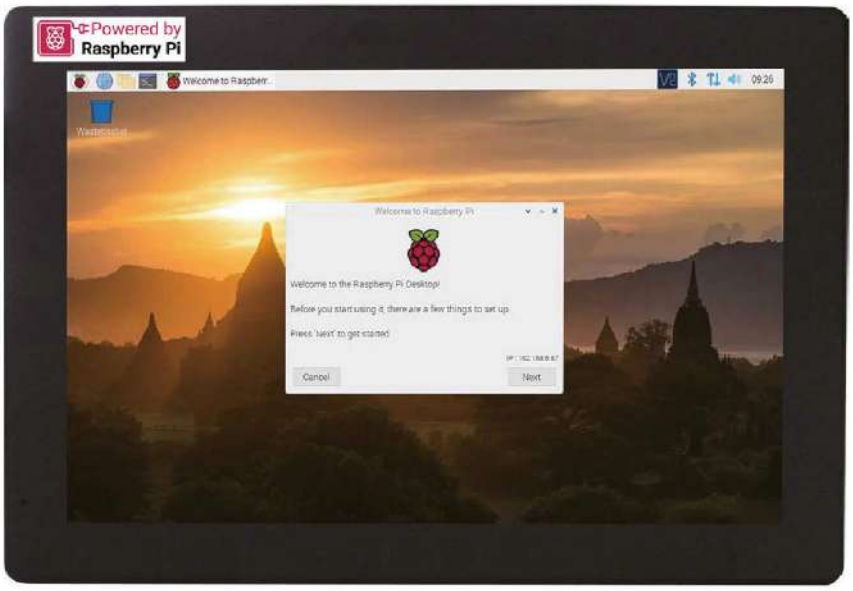
listing2.py

DOWNLOAD THE FULL CODE: magpi.cc/seancode

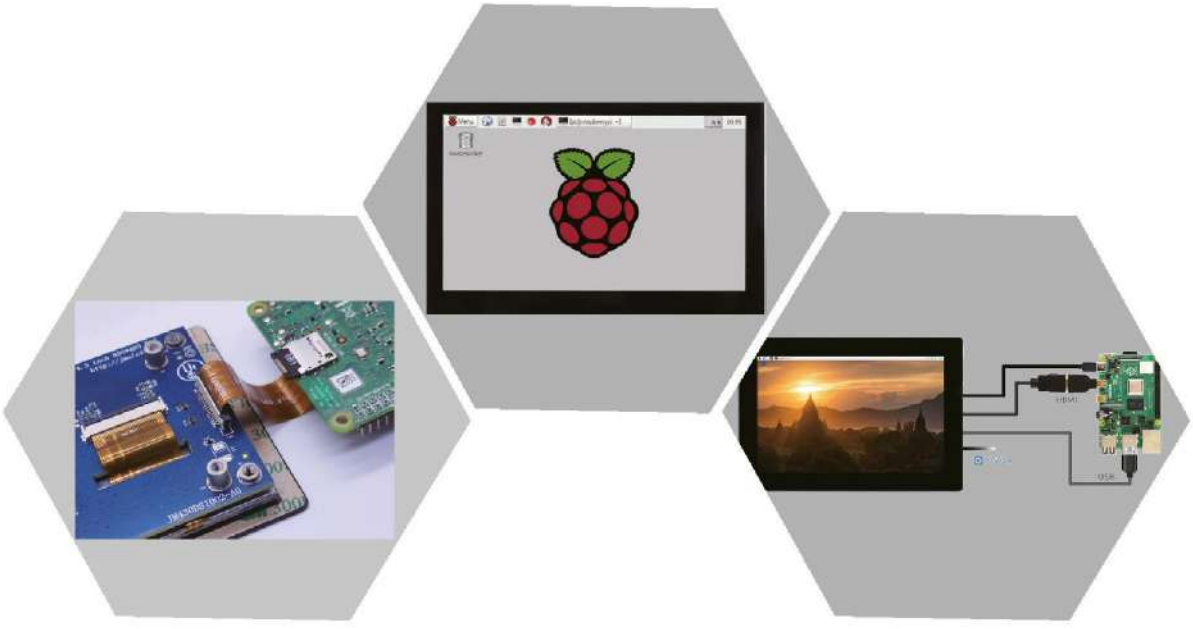
> Language: **Python**

```
# Download web pages into a docx file
import requests, sys
from bs4 import BeautifulSoup
from docx import Document

print("Paste in the URLs (Ctrl-D to end input): ")
urls = sys.stdin.readlines()
urls = [url.strip() for url in urls]
filename = "output.docx"
doc = Document()

for source_number, url in enumerate(urls):
    print(f"Fetching {url}")
    response = requests.get(url)
    content = response.content
    soup = BeautifulSoup(content, "html.parser")
    for remove_me in soup.find_all(["nav", "footer"]):
        remove_me.extract()

    doc.add_heading(f"{source_number + 1} - {url}", 1)
    title = soup.title.string
    doc.add_heading(f"{source_number + 1} - {title}", 0)
    for part in soup.find_all(["p", "h1", "h2", "h3", "h4", "h5", "h6", "table", "li", "blockquote"]):
        if part.name in ["h1", "h2", "h3"]:
            doc.add_heading(part.text, 2)
        elif part.name == "li":
            doc.add_paragraph(part.text, style="List Bullet")
        elif part.text:
            doc.add_paragraph(part.text)
    doc.add_page_break()
doc.save(filename)
print(f"Saved as {filename}")
```

Raspberry Pi Custom DSI Display
CM4 Carryboard, and Industrial Tablet



Custom DSI Displays



Development of
CM4-compatible carryboard

Industrial Tablet
development capabilities

Looking for cooperation partners around the world.

Contact us
Tel: +44 7587611960
E-mail: straight@jmolcd.com
Website: www.jmolcd.com

Get started with Raspberry Pi AI Kit

Place a Hailo AI module inside the Raspberry Pi M.2 HAT+ to give a Raspberry Pi accelerated high-performance AI

MAKER

Nate Contino

Nate is a retrofuturist and writes documentation for Raspberry Pi.

magpi.cc/natec

The Raspberry Pi AI Kit bundles the Raspberry Pi M.2 HAT+ with a Hailo AI acceleration module for use with Raspberry Pi 5. The kit contains the Hailo AI module containing a neural processing unit (NPU), a Raspberry Pi M.2 HAT+, to connect the AI module to your Raspberry Pi 5, a thermal pad pre-fitted between the module and the M.2 HAT+, a mounting hardware kit, and a 16mm stacking GPIO header

The AI module features a 13 tera-operations per second (TOPS) neural network inference accelerator built around the Hailo-8L chip in an M.2 2242 form factor.

In this tutorial we will look at how to assemble and attach AI Kit to Raspberry Pi, and use it for high-performance AI tasks.

To use the AI Kit, you will need a Raspberry Pi 5. Each AI Kit comes with a pre-installed AI module, ribbon cable, GPIO stacking header, and mounting hardware. Complete the following instructions to install your AI Kit:

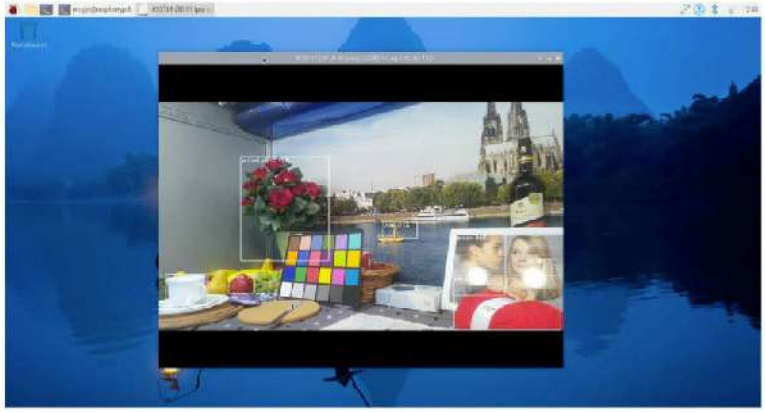
First, ensure that your Raspberry Pi runs the latest software. Run the following command to update:

```
$ sudo apt update && sudo apt full-upgrade
```

Next, ensure that your Raspberry Pi firmware is up-to-date (**magpi.cc/updatebootloader**). Run the following command to see what firmware you're running:

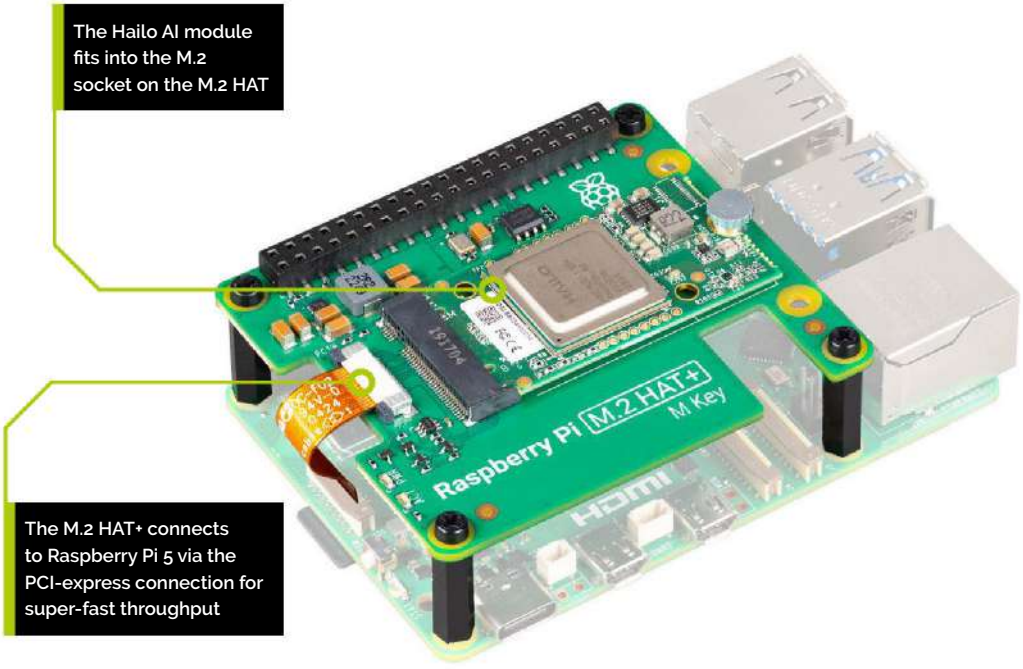
```
$ sudo rpi-eeprom-update
```

If you see 6 December 2023 or a later



▼ An object detection model running on AI Kit

▲ A still life in our Image Lab for testing the Camera Module



▼ Install the Active Cooler first

You'll Need

- Raspberry Pi 5 **magpi.cc/raspberrypi5**
- Raspberry Pi OS (64-bit) 'Bookworm'
- AI Kit **magpi.cc/aikit**
- Camera Module (optional) **magpi.cc/cameramodule**

Warning!
Turn off power

Always disconnect your Raspberry Pi from power before connecting or disconnecting a device from the M.2 slot.

magpi.cc/power

date, proceed to the next step. If you see a date earlier than 6 December 2023, run the following command to open the Raspberry Pi Configuration CLI:

```
$ sudo raspi-config
```

Under Advanced Options > Bootloader Version, choose Latest. Then, exit raspi-config with Finish or the **ESC** key.

Run the following command to update your firmware to the latest version:

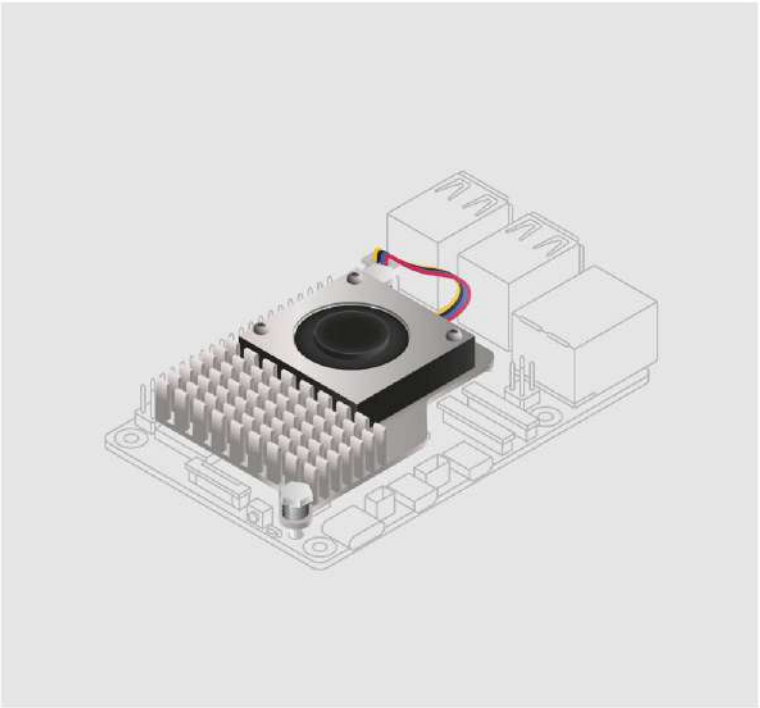
```
$ sudo rpi-eeprom-update -a
```

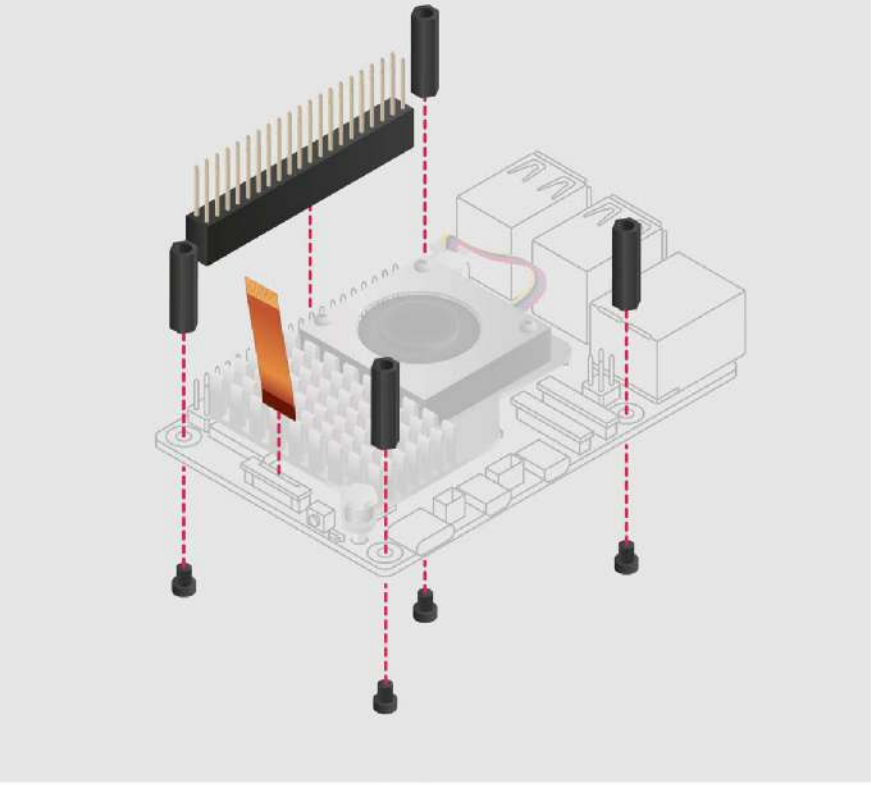
Then, reboot with `sudo reboot`.

Installing the AI Kit

Disconnect the Raspberry Pi from power before beginning installation. For the best performance, we recommend using the AI Kit with the Raspberry Pi Active Cooler. If you have an Active Cooler, install it before installing the AI Kit.

Install the spacers using four of the provided





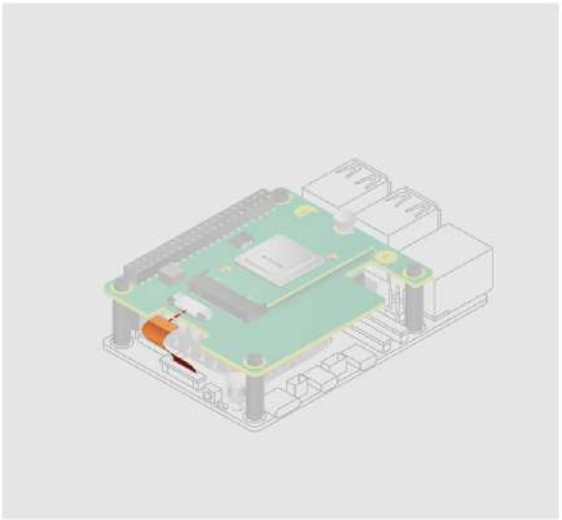
Attach the stand-offs and GPIO Pin header

The ribbon cable connects AI Kit to Raspberry Pi 5

screws. Firmly press the GPIO stacking header on top of the Raspberry Pi GPIO pins; orientation does not matter as long as all pins fit into place. Disconnect the ribbon cable from the AI Kit, and insert the other end into the PCIe port of your Raspberry Pi. Lift the ribbon cable holder from both sides, then insert the cable with the copper contact points facing inward, towards the USB ports. With the ribbon cable fully and evenly inserted into the PCIe port, push the cable holder down from both sides to secure the ribbon cable firmly in place.

Set the AI Kit on top of the spacers, and use the four remaining screws to secure it in place.

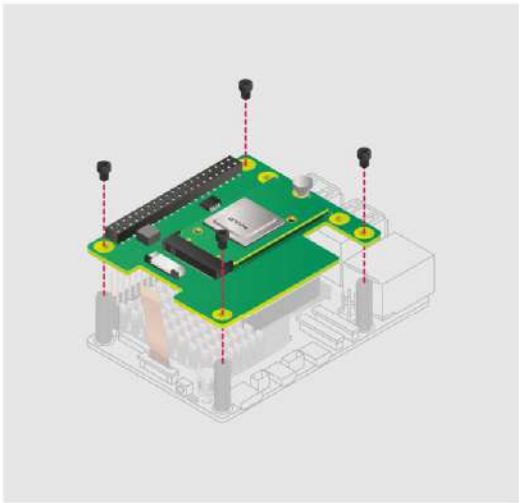
Insert the ribbon cable into the slot on the AI Kit. Lift the ribbon cable holder from both sides, then insert the cable with the copper contact points facing up. With the ribbon cable fully and evenly inserted into the port, push the cable holder down from both sides to secure the ribbon cable firmly in place.

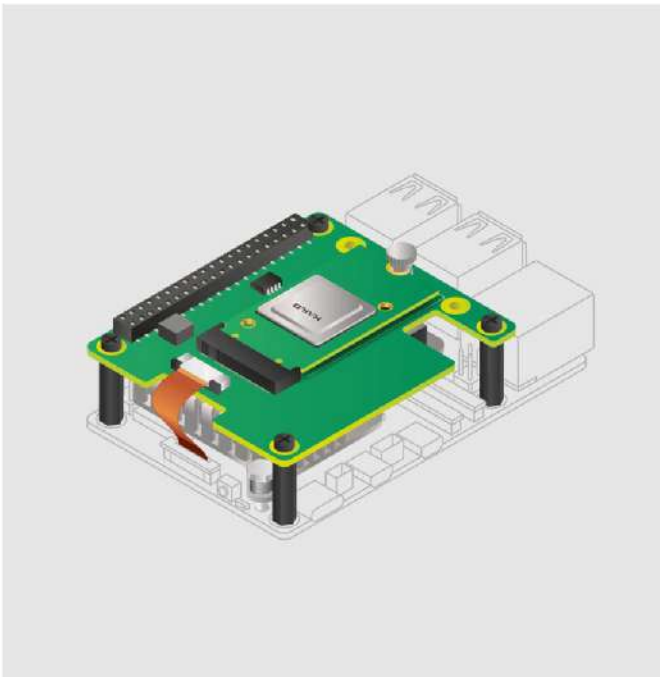


Congratulations, you have successfully installed the AI Kit. Connect your Raspberry Pi to power; Raspberry Pi OS will automatically detect the AI Kit.

Use screws to attach the AI Kit board and secure it in place

Make sure everything is securely connected





Getting started

This section will help you set up the Raspberry Pi AI Kit with your Raspberry Pi 5. This will enable you to run rpicam-apps camera demos using the Hailo AI neural network accelerator.

Attach the camera to your Raspberry Pi 5 board following the instructions at Install a Raspberry Pi Camera (**magpi.cc/camdocs**). You can skip reconnecting your Raspberry Pi to power, because you'll need to disconnect your Raspberry Pi from power for the next step.

> Once everything is installed correctly, it's time to run some demos

Follow the instructions to enable PCIe Gen 3.0 (**magpi.cc/pcigen3**). This step is optional, but highly recommended to achieve the best performance with your AI Kit.

```
$ sudo raspi-config
```

Complete the following steps to enable PCIe Gen 3.0 speeds: Select Advanced Options. PCIe Speed and Choose YES to enable PCIe Gen 3 mode. Select FINISH to exit and choose YES to "Would you like to reboot now?".

Install the dependencies required to use the AI Kit. Run the following command from a terminal window:

```
$ sudo apt install hailo-all
```

This installs the following dependencies:

- Hailo kernel device driver and firmware
- HailoRT middleware software
- Hailo Tappas core post-processing libraries
- The rpicam-apps Hailo post-processing software demo stages

Finally, reboot your Raspberry Pi with sudo reboot for these settings to take effect. To ensure everything is running correctly, run the following command:

```
$ hailortcli fw-control identify
```

If you see output similar to the following, you've successfully installed the AI Kit and its software dependencies:

```
Executing on device: 0000:01:00.0
Identifying board
Control Protocol Version: 2
Firmware Version: 4.17.0
(release,app,extended context switch buffer)
Logger Version: 0
Board Name: Hailo-8
Device Architecture: HAILO8L
Serial Number: HLDDLBB234500054
Part Number: HM21LB1C2LAE
Product Name: HAILO-8L AI ACC M.2 B+M KEY
MODULE EXT TMP
```

Additionally, you can run `dmesg | grep -i hailo` to check the kernel logs, which should yield output similar to the following:

```
[    3.049657] hailo: Init module. driver
version 4.17.0
...
[    3.231845] hailo 0000:01:00.0: Probing:
Added board 1e60-2864, /dev/hailo0
```

To ensure the camera is operating correctly, run the following command:

```
$ rpicam-hello -t 10s
```

This starts the camera and shows a preview window for ten seconds. Once you have verified everything is installed correctly, it's time to run some demos.

Run the demos

The rpicam-apps suite of camera applications implements a post-processing framework (**magpi.cc/postproc**). This section contains a few demo post-processing stages that highlight some of the capabilities of the AI Kit.

The following demos use rpicam-hello, which by default displays a preview window. However, you can use other rpicam-apps instead, including rpicam-vid and rpicam-still. You may need

Top Tip

rpicam-apps

The commands provided use the JSON files in this repository. To make it easy to reference these files, this command creates the cloned rpicam-apps directory in your home folder. If you modify the location of this directory, you must also alter the demo commands to reference the new location of the JSON files.

to add or modify some command line options to make the demo commands compatible with alternative applications.

To begin, download the post-processing JSON files required for the demos. These files determine which post-processing stages to run and configure the behaviour of each stage. For example, you can enable, disable, strengthen, or weaken the strength of the temporal filtering in the object detection demos. Or you could enable or disable the output mask drawing in the segmentation demo.

To download the entire collection of post-processing JSON files, clone the rpicam-apps repo. Run the following command to clone only the most recent commit from the repo, saving space:

```
$ git clone --depth 1 https://github.com/
raspberrypi/rpicam-apps.git ~/rpicam-apps
```

Object Detection

This demo displays bounding boxes around objects detected by a neural network. To disable the viewfinder, use the `-n` flag. To return purely textual output describing the objects detected, add the `-v 2` option. Run the following command to try the demo on your Raspberry Pi:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolov6_inference.
json --lores-width 640 --lores-height 640
```

Alternatively, you can try another model with different trade-offs in performance and efficiency.

To run the demo with the Yolov8 model, run the following command:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolov8_inference.
json --lores-width 640 --lores-height 640
```

To run the demo with the YoloX model, run the following command:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolox_inference.json
--lores-width 640 --lores-height 640
```

> Hailo's extensive model zoo contains a large number of neural networks

To run the demo with the Yolov5 Person and Face model, run the following command:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolov5_personface.
json --lores-width 640 --lores-height 640
```

Image Segmentation

This demo performs object detection and segments the object by drawing a colour mask on the viewfinder image. Run the following command to try the demo on your Raspberry Pi:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolov5_segmentation.
json --lores-width 640 --lores-height 640
--framerate 20
```

Pose Estimation

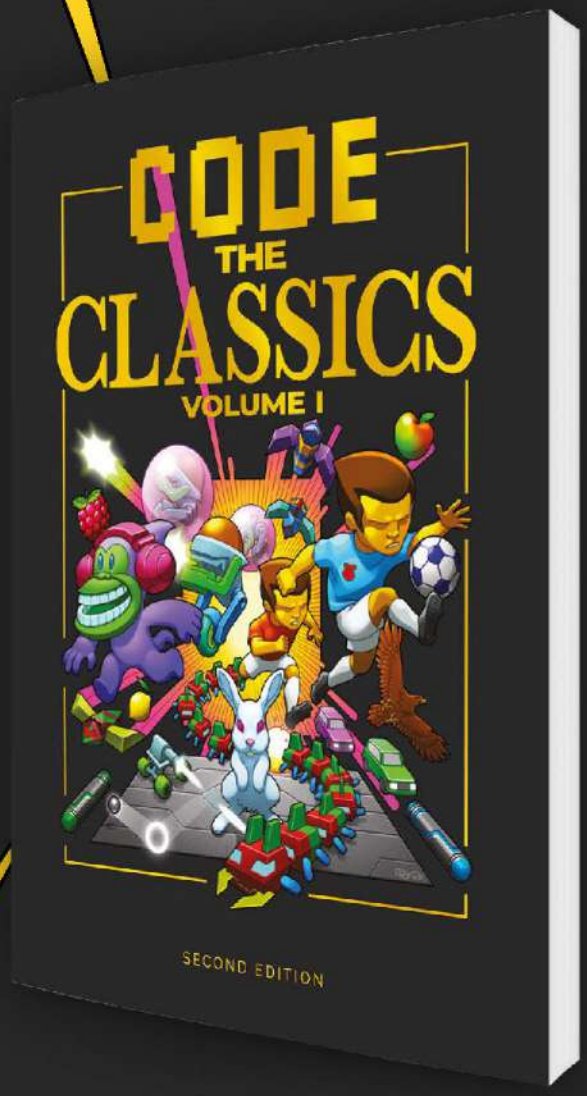
This demo performs 17-point human pose estimation, drawing lines connecting the detected points. Run the following command to try the demo on your Raspberry Pi:

```
$ rpicam-hello -t 0 --post-process-file ~/
rpicam-apps/assets/hailo_yolov8_pose.json
--lores-width 640 --lores-height 640
```

Hailo has also created a set of demos that you can run on a Raspberry Pi 5, available in the hailo-ai/hailo-rpi5-examples GitHub repository (**magpi.cc/hailorpi5**).

You can find Hailo's extensive model zoo, which contains a large number of neural networks, in the hailo-ai/hailo_model_zoo GitHub repository (**magpi.cc/hailozoo**).

Check out the Hailo community forums and developer zone (community.hailo.ai) for further discussions on the Hailo hardware and tooling.



New in the Second Edition

- ***Expanded Python and Pygame Zero tutorials***
- *A GitHub tutorial for working with example code*
- ***Bug fixes and other improvements***

This stunning 240-page hardback book not only tells the stories of some of the seminal video games of the 1970s and 1980s, but shows you how to create your own games inspired by them using Python and Pygame Zero, following examples programmed by Raspberry Pi founder Eben Upton.

Available now: **magpi.cc/store**

Part 01

Build a private cloud server

Fed up with service subscriptions? Want to do things yourself and protect your privacy? This new series shows you how

MAKER

PJ Evans

PJ is a writer, software engineer and home automation enthusiast. He is guaranteed 100% AI-free, even if his home server isn't.

@mrpjevans

We live in a golden age of free open-source software. We also live in an age of megacorps offering 'free' services that routinely sell your data via arcane terms and conditions. Luckily there is a new appetite for running your own services and regaining control thanks to fast and affordable bandwidth and the growth of small yet powerful computers to do the work (sound familiar?). In this new series, we're going to go step-by-step through the process of building a reliable cloud server that you own and control. We'll cover file sharing, email, social media and other collaboration tools. Let's start by choosing and building our base server.

You'll Need

- Argon EON NAS enclosure (or similar) **magpi.cc/eonnas**
- At least two HDD drives
- Additional OS drive (ideally SSD)
- Fast internet in both directions
- SATA to USB3 cable (optional)

01 Is this a good idea?

We're setting out to build a server that can replicate many of the popular 'SaaS' (software as a service) solutions out there, from email to collaborative spreadsheeting. This is only practical if you have the time to devote to keeping it updated (security especially) and are happy to deal with the inevitable glitches. You'll also need a decent internet connection if you want to run things like an email server or a Mastodon instance. There are many reasons to undertake this project, some political, some practical, but we're mainly doing it because it's a fun learning experience.

02 Choose your storage

We're making file storage an integral part of the server. As such, we want lots of space and some form of backup strategy. Although you can always upgrade your storage as you go, it's best to think ahead and get as large a solution as you can afford. Our project is using 2 × 4TB SATA hard drives as file storage, plus a 250GB SSD SATA drive for the operating system. We'd happily go SSD all the way, but the costs are prohibitive. Separating the OS from the file storage allows us to use a fast SSD and makes it easier to upgrade in future. You could also use a microSD card for the OS.

03 Prepare the operating system

We'll refer to our choice of hardware from now on, so alter the process as needed for your setup. We're using a small but fast SSD for our operating system. This means we need to copy the image directly onto the SSD. The easiest way to do this is to use a cheap SATA to USB3 cable on your main computer or other Raspberry Pi and use Raspberry Pi Imager (**magpi.cc/imager**), selecting 'Raspberry Pi Lite (64-bit)'. If you don't have a cable, install the drive in the NAS and boot off a microSD card and use that to burn the OS onto the SSD. Either way, use the customisation feature to choose a good network name and enable SSH for remote access.

04 Assemble your NAS

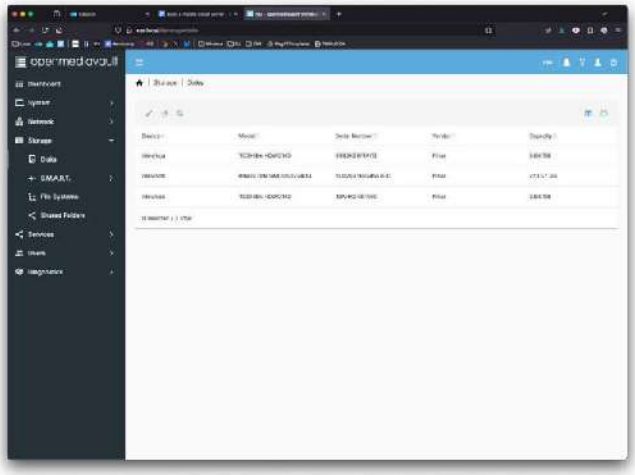
This will vary depending on what type of disk setup and enclosure you are using, but either way now is the time to install your disks. The Argon EON requires some initial disassembly to access the space for a 4GB Raspberry Pi 4. For our setup we have 4 SATA slots available. The first (left from the front-on view) will host our SSD drive (a quirk of the EON, due to its shape, is that slots 1 and 4 must be smaller drives). Next we install our main file storage drives in slots 2 and 3. All drives must be secured to the backplate with the supplied screws. Finally the EON has a USB 3 coupler to connect the disks to the Raspberry Pi.

> File sharing, email, social media and other tools

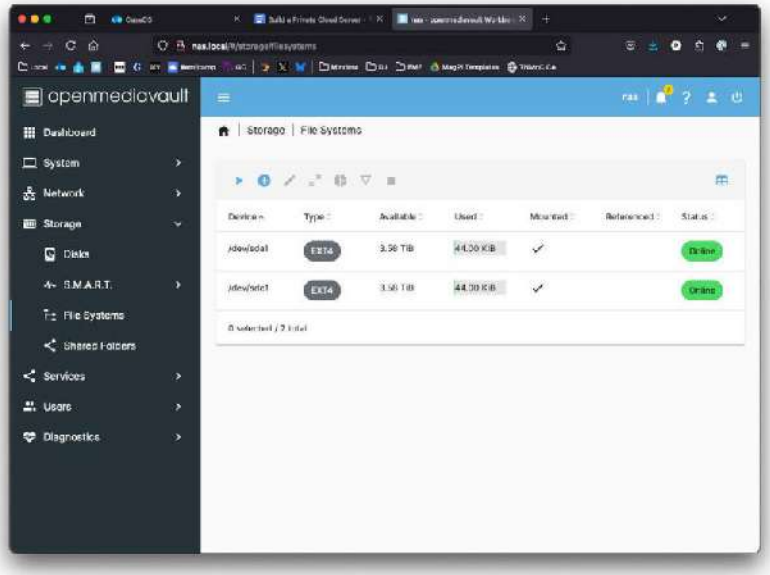
Top Tip

Choosing disks

If using hard disk drives, consider buying dedicated NAS drives. These tend to be slower than desktop drives but much more efficient and quieter.



When you first log in to OpenMediaVault you'll see your disks listed as devices



You can create, view and manage file systems on this page

05 Test booting

Time to fire up the server for the first time and check everything is in order. It's worth connecting a monitor and keyboard just in case something is out of place. If you find the SSD doesn't boot, it may be because the Raspberry Pi isn't checking for it and you need to change the bootloader. Open Raspberry Pi Imager. Under operating systems choose 'Misc utility images' > 'Bootloader' > 'SD Card Boot', burn that to a microSD card and boot your NAS from it. When a green screen appears, power off and remove the card. Your SSD should now boot normally.

06 Wire up and update

We strongly recommend using a wired connection for this project as it may become essential for reliable performance, especially if you're using email or social media services. You'll also need to check in on the server from time to time, so make sure you can use SSH to log in remotely. You should be able to reach the NAS using your choice of name plus '.local'. So, if you set the hostname to 'nas' in Raspberry Pi Imager, you can SSH in using 'nas.local' (note: not all routers support this feature). Once logged in make sure everything is up-to-date by running the following from the command-line:

```
sudo apt -y update && sudo apt -y upgrade
```

If you're using the Argon EON, you should also install its config scripts at this point. See the manual for details.

07 Install OpenMediaVault

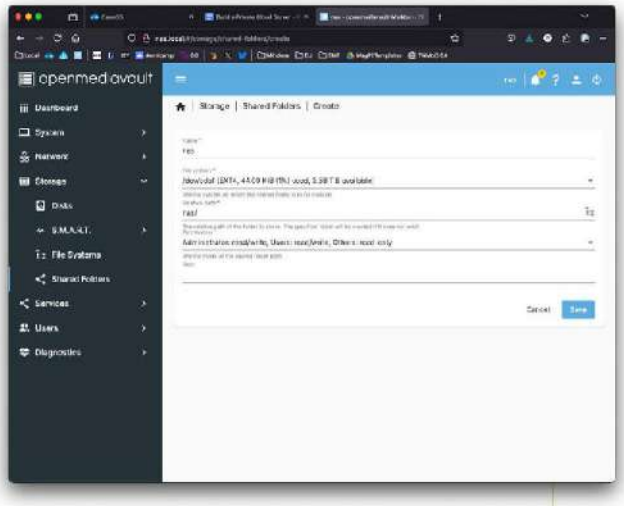
OpenMediaVault (or OMV) is a robust, well-featured framework for creating our own home storage. It's excellent for avoiding some of the more scary command line utilities for handling disk storage and setting up sharing. OMV's job is to help you prepare your disks and share them securely with the other computers on your network. The OMV team have provided a custom setup script at and you can run it directly by entering:

```
wget -O - https://raw.githubusercontent.
com/OpenMediaVault-Plugin-Developers/
installScript/master/install | sudo bash
```

There will be a lot of text and the process will go on for several minutes. At the end the server will reboot. Once complete you should be able to access OMV in a web browser at **http://nas.local/** (or your equivalent). You can login with username 'admin' and password 'openmediavault'. Remember to change that password!

08 Configure your storage

You can now see the OMV web interface for the first time. Feel free to have an exploration. Assuming you're starting with new disks, you need to partition and format them. Luckily OMV makes this a much easier process than using the command line. Click on 'Storage' then 'Disks'. You'll see your disks, including the OS disk, listed. If not starting afresh you can wipe them here. To actually use the disks, click on 'File Systems' and



The final step is to create share points on each disk (in reality each is a root folder on the disk) and then share using SMB

the plus sign on the toolbar. Select 'EXT4' and choose your first disk. It may take a few minutes to complete. Repeat for each disk.

09 Mounting and sharing

Once each disk has formatted you are presented with a screen to mount the drive. Select the disk you just formatted and click 'Mount'. The file system is now available to OMV. You'll see a yellow 'Pending' prompt. This is a feature of OMV. Wherever you make changes, they will not take effect until you acknowledge that prompt. Now click 'Shared Folders' and the plus button. Give your share a meaningful name (we used 'nas') and choose your first disk. Click 'Save' and apply the changes. Now click 'Services', 'SMB/CIFS' then 'Settings' and enabled the service. Now go to 'Shares' then the plus button and select your shared folder. Save and apply.

10 Connect to your new share

In the previous steps you formatted your disks, created a shared folder on one of them and then allowed access using the popular SMB protocol, which is supported by a wide range of operating systems. You can now connect to your new storage! In each case, use your Raspberry Pi OS username and password as credentials. In Windows right-click on 'This PC' and select 'Map network drive'. Enter the details in format **\\nas.local\nas** (change for your setup) and select 'use different credentials'. In macOS, select Finder then 'Go' from the menu and 'Connect to server'. Enter the address as '**smb://nas.local/nas**' and connect.

11 Nightly backup

So what about the other disk? We're going to use that to back up our precious data. Under 'Storage', 'Shared Folders' create a share for the second disk (e.g. '**nas-backup**'). Then under 'Services' select 'Rsync' then 'Tasks'. Select your first share as the source and the second share as the target. Change 'minute' to 0 and 'hour' to 0, leaving all other options unchanged. Save and apply. Now, every night at midnight, the contents of the primary disk will be backed up to the secondary disk. Although there is a small window where a file does not have an instant backup (we'll address this, and off-site backups, in the future) a deleted file will be instantly recoverable.

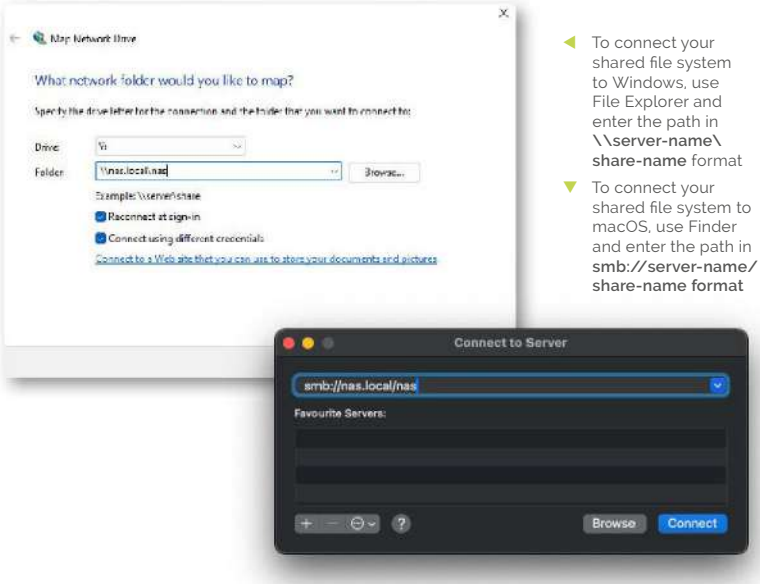
12 More with OpenMediaVault

We have barely scraped the surface of what OpenMediaVault can do. Make sure you explore the menus and look for other features such as email alerting, configuring disks to run quieter and, importantly, using S.M.A.R.T. to monitor the health of your disks. You now have file sharing available across your home network that you can share with your housemates or family. You can even create individual secured folders for personal use. Next time we'll look at accessing that data securely from anywhere on the planet, the first step in creating a true private cloud server.

Top Tip

What about RAID?

We've not used RAID (Redundant Array of Inexpensive Disks) in our setup as we're prioritising backup over availability. RAID is not a backup solution – if you delete a file on one disk, it's immediately gone from the other!



To connect your shared file system to Windows, use File Explorer and enter the path in **\\server-name\share-name** format

To connect your shared file system to macOS, use Finder and enter the path in **smb://server-name/share-name format**

Capture images and video with Raspberry Pi Camera Module

Learn how to capture images and video using rpicam, the new command-line application for Raspberry Pi

MAKER

David Plowman

David is an engineer at Raspberry Pi with a special interest in camera software and algorithms, plus image processing hardware.

magpi.cc

The latest version Raspberry Pi OS (based on Bookworm) has a new camera system called rpicam. This replaces the older libcamera application.

Raspberry Pi has made five different types of cameras, of which all but the original v1 camera is currently in production. Some of them come in two or more different variants.

The official Raspberry Pi camera modules are:

- Raspberry Pi Camera Module
- Raspberry Pi Camera Module 2
- Raspberry Pi High Quality Camera
- Raspberry Pi Camera Module 3
- Raspberry Pi Global Shutter Camera

First, you will need to set up your Camera Module by attaching it to your Raspberry Pi. See The MagPi #129 (**magpi.cc/129**) for our guide to connecting Raspberry Pi Camera Module to your Raspberry Pi board.

Once that's set up, we can start looking at rpicam. To test that your camera is correctly connected and working, you'll need to install Raspberry Pi OS onto your board. Using Raspberry Pi Imager (**magpi.cc/imager**), write the default option, Raspberry Pi OS (64-bit) to a microSD card.

Insert this microSD card into your Raspberry Pi and power it on. Let it perform all the necessary updates when it first boots and then finally reboot it again.

For those less familiar with Raspberry Pi computers, we would recommend performing your initial testing with a keyboard and screen attached directly to your Raspberry Pi. Using the Raspberry menu, open the Accessories category and click on the Terminal button. Into the Terminal window it should be sufficient to type:

```
rpicam-hello
```

This will open a camera preview window and display the camera images there for about five seconds.

Capturing your first photo

Raspberry Pi OS comes with a number of command line utilities for capturing images in various different ways. We've already seen `rpicam-hello` which we used to test that the camera was working. The next one is `rpicam-still`. This opens up a camera preview window just like `rpicam-hello` did but once it's run for five seconds it will capture a high resolution still photograph as a JPEG file, and then quit. To use it like this, enter:

```
rpicam-still -o picture.jpg
```

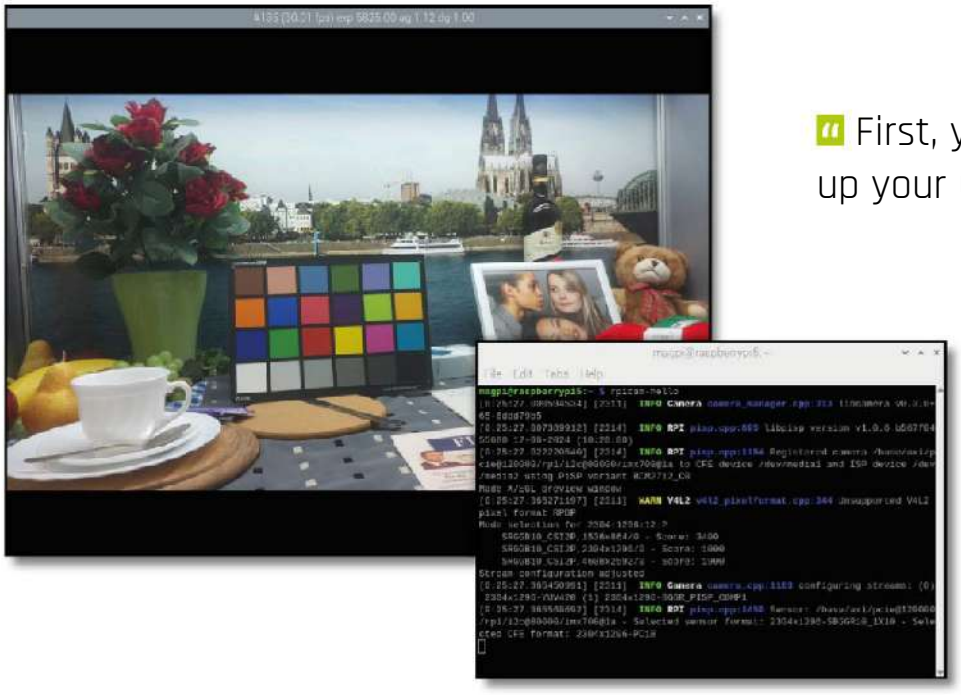
Note how the `-o` option specifies the name of the file to which the image is saved. There is also the `-t` option which specifies how long, in milliseconds, the preview window runs before the

By default, each camera will capture images at the maximum resolution. It is possible to adjust the capture resolution in software

The High Quality Camera Module is focused manually, the new Camera Module 3 has autofocus that can be controlled in software

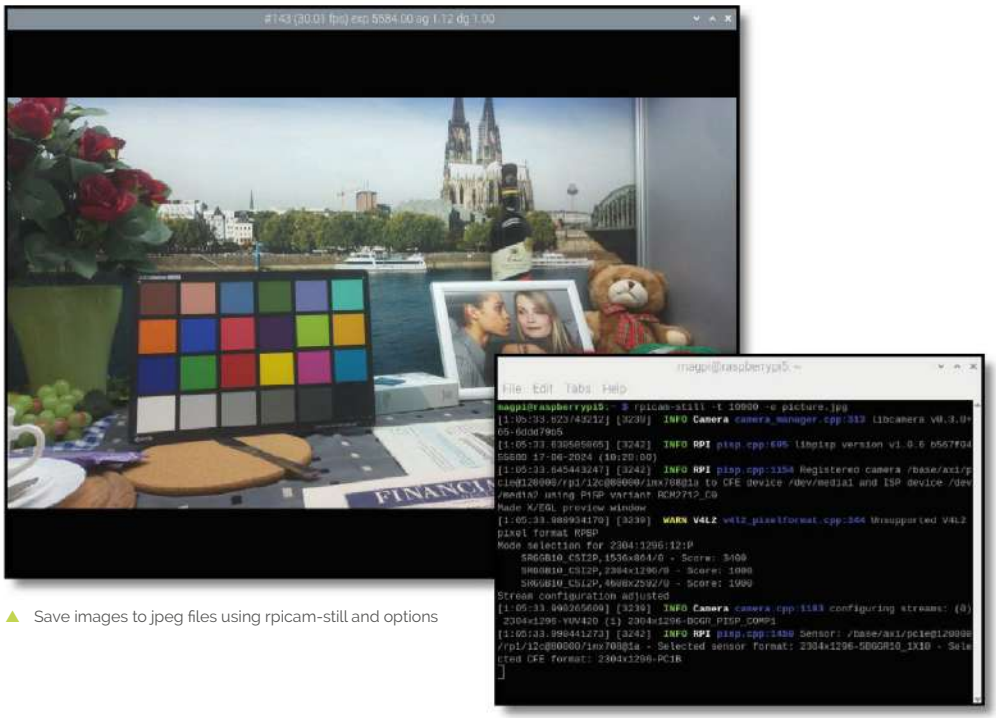
You'll Need

- Raspberry Pi
- Camera Module **magpi.cc/cameramodule**
- Raspberry Pi OS (64-bit) **magpi.cc/raspberrypios**



> First, you will need to set up your Camera Module

Test your camera is functioning correctly with the rpicam-hello Terminal command



▲ Save images to jpeg files using rpicam-still and options

image is captured. So to capture the image after ten seconds of preview, use:

```
rpicam-still -t 10000 -o picture.jpg
```

You can view your pictures by double-clicking on the files you've saved in the File Manager.

Capturing your first video

There's another command line tool for capturing videos, this time it's called `rpicam-vid`. You can use the `-o` and `-t` options as you did before, though this time `-t` specifies how long (again in milliseconds) the video lasts. To save a ten-second video, use:

```
rpicam-vid -t 10000 -o video.h264
```

Note that the file name ends with .h264 which identifies the format of the video file. Unfortunately not all video players will play the .h264 format correctly, so we would recommend using a utility like ffplay. To play your video file with ffplay, type:

```
ffplay video.h264
```

We've already met `rpicam-still` which allows us to capture still images. But it also has many more options controlling how it captures images, and the file formats and available image encoders. We'll discover some of those in this tutorial. For now, we shall continue to assume that you have your keyboard and a monitor plugged directly into your Raspberry Pi.

Resolutions and formats

By default, `rpicam-still` will capture images at the maximum available resolution supported by the camera. For the HQ camera, for example, this means images that are 4056×3040 pixels in size. But it's easy to change this with the `--width` and `--height` parameters. For example, if you want to capture an image that is 1536×1024 pixels instead, you should use:

```
rpicam-still --width 1536 --height 1024 -o
smaller.jpg
```

Images are normally encoded in some way (often they are compressed so that they aren't so large) and then saved to a standard file format on the disk. There are several of these formats:

JPEG files

The default file format used by `rpicam-still` is JPEG because of the very wide support that it has, combined with a useful level of compression. Usually, JPEG files are given the extension `.jpg`, or sometimes `.jpeg`, as we have already seen.

JPEG is a lossy compression format, meaning that the result of loading your JPEG file is no longer identical to the image that you started from, though the differences are arranged to be ones that you will least notice. In return, you get

to specify a quality factor indicating how much compression you want. At very low values your image will indeed deteriorate, but at very high values the JPEG process will be visually lossless, that is, it really will look as good as the original camera image.

The quality factor ranges from 1 to 99, and if you don't give one it will choose the value 93, which equates informally to "quite high quality". To save a JPEG at a different quality level, use the `-q` (or equivalently `--quality`) parameter like this:

```
rpicam-still -q 80 -o test.jpg
```

Please experiment with quality values like 10 and 50 to see what you get!

PNG files

PNG (or 'Portable Network Graphic') is another file and compression format which enjoys wide support. PNG is a lossless compression format, meaning that you are guaranteed to be able to recover exactly the same image that you started from (unlike JPEG). On the downside, PNG files are normally larger than JPEG files, and they are harder work to create, which takes longer.

To save a PNG file you'll need to tell `rpicam-still` that you want to use a different encoder using the `-e` or `--encoder` parameter, like this:

```
rpicam-still -e png -o test.png
```

Note that it's necessary to specify the `-e` option to get a PNG file – changing the file name on its own is not enough. PNG does not support a quality parameter.

DNG files

DNG, or 'Digital Negative', files are quite different from PNG files despite the similar abbreviation! As the full name suggests, they're somewhat analogous to the 'negatives' we had from film cameras before developing them into photos. In our case, the DNG file stores the raw numbers received from the image sensor before the hardware on Raspberry Pi 'develops' it into a viewable (JPEG or PNG) image. DNG files are saved alongside the 'developed' JPEG or PNG version of the same image. To save a DNG file, use the `-r` (or `--raw`) parameter:

```
rpicam-still -r on test.jpg
```

`rpicam-still` will automatically replace `.jpg` by `.dng` in the DNG file name, giving both `test.dng` and `test.jpg` in this case.

Many third-party software programs exist to 'develop' these DNG files interactively – a rather complex procedure beyond the scope of this guide.

Time-lapse captures

A time-lapse capture is where we capture an image at regular intervals, perhaps every minute, hour or day, and then reassemble them into a video where we play them back at a much faster rate. `rpicam-still` is all set up to capture the images we need out of the box.

We need to introduce the `--timelapse` option and we'll review a couple that we've seen before:

- `-t` or `--timeout` The length of time in milliseconds for which to perform a capture. In the case of timelapse, `rpicam-still` will run capturing images in total for this duration. You can pass the value 0 which means "run indefinitely" (you will have to stop `rpicam-still` manually, for example by pressing **CTRL**+**C** on the keyboard).
- `--timelapse` The length of time, again in milliseconds, between each of the timelapse captures.
- `-o` or `--output` The name of the output file or files. For timelapse captures, we can't give all the images the same name so we use a special syntax that includes an image counter. For example, `-o capture_%04d.jpg` means that all the files are named `capture_`, followed by a counter, and then `.jpg. %04d` specifies how the counter is formatted, in this case the 0 means to add leading zeroes to the number and the 4 means "so that every number has at least 4 digits". This is useful so that listing your image files will return the list to you in chronological order.

Top Tip

libcamera to rpicam

Raspberry Pi OS Bookworm renamed the camera capture applications from libcamera-* to rpicam-*. Symbolic links allow users to use the old names for now. Adopt the new application names as soon as possible. Raspberry Pi OS versions prior to Bookworm still use the libcamera-* name.

▲ Video files can be recorded using rpicam-vid and played back using the ffmpeg command

> When using the Camera Module 3, autofocus is enabled automatically

Let's try an example:

```
rpicam-still -o capture_%04d.jpg --timelapse
5000 -t 30000 --width 1024 --height 768
```

This will run for 30 seconds, capturing an image every five seconds, and they'll be called `capture_0000.jpg` through to `capture_0004.jpg`. Note how we've reduced the resolution of the images to something that is more appropriate to the final video that we want to create.

Assembling your images into a video

There's a handy tool called FFmpeg which is capable of turning your sequence of still images into a video. We can use it like this:

```
ffmpeg -r 2 -i capture_%04d.jpg video.mp4
```

Note how we format the name of the input files with the special `%` syntax in the same way as we did for `libcamera-still`. The `-r` parameter gives the framerate of the output video, which is two frames per second in this case. We've chosen the output file to have the MP4 format (which is generally well supported), and called it `video.mp4`.

FFmpeg is a highly versatile tool that we recommend learning about.

Capturing when a key is pressed

Rather than doing regular captures, we can also do them in response to an event, or a key press. There are a couple of new option parameters to learn.

- `--datetime` Use this instead of `-o` to name the output file after the current date and time. The format will be `MMDDhhmmss.jpg` where `MM` and `DD` are the month and date number, and `hh`, `mm` and `ss` are hours, minutes and seconds.
- `-k` or `--keypress` Capture an image when **ENTER** is pressed on the keyboard. Press **X** and press **ENTER** to quit.

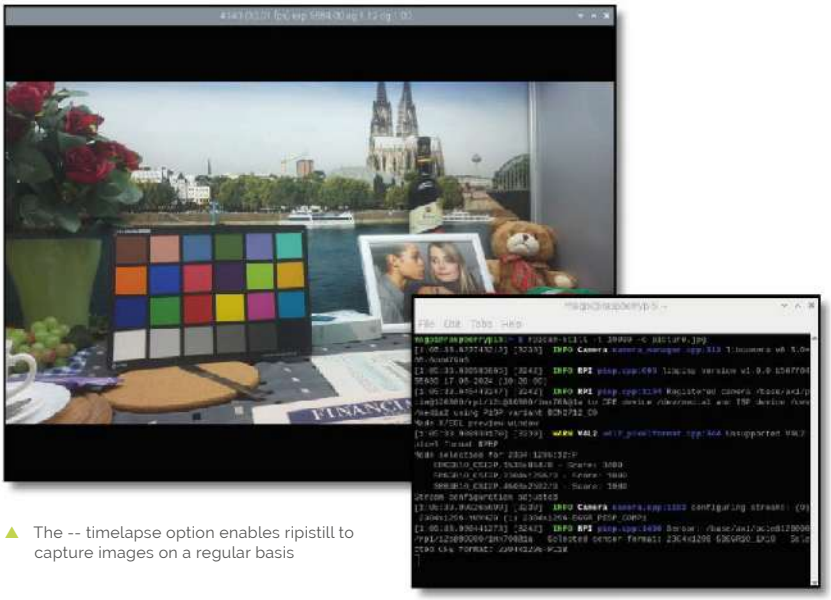
So the command to use is this:

```
rpicam-still -t 0 --keypress --datetime
```

Here we're running the capture indefinitely, so we'll have to type **X** followed by **ENTER** to quit (or press **CTRL+C**). Files would have names like `0405102742.jpg`, meaning '10:27am, and 42 seconds, on 5 April'.

Capturing in response to a signal

For those familiar with Linux signals, an alternative to pressing a key is to send a signal instead. To do this, simply use `-s` or `--signal` instead of `-k` (or `--keypress`).



▲ The -- timelapse option enables ripistill to capture images on a regular basis

Top Tip

Compatability

Raspberry Pi Camera Modules are compatible with all Raspberry Pi computers with CSI connectors - that is, all models except Raspberry Pi 400 and the 2016 launch version of Zero.

To send a capture signal to `libcamera-still`, first start it and then type the following into another terminal window:

```
kill -SIGUSR1 `pidof libcamera-still`
```

And you can force `libcamera-still` to quit with

```
kill -SIGUSR2 `pidof libcamera-still`
```

Autofocus and High Dynamic Range

Autofocus and High Dynamic Range imaging are supported only on Raspberry Pi Camera Module 3 at the time of writing.

When using the Camera Module 3, autofocus is enabled automatically in continuous mode. This means that the camera lens will move whenever necessary to maintain optimal focus on the centre part of the image, and this is probably what most users will want most of the time.

It's also possible to turn off autofocus and set the focus position of the lens manually. To do this, use the `--lens-position` parameter, and pass it a value measured in dioptres, meaning the reciprocal of the focus distance. Thus, to focus at a very close distance of about 0.1m, pass in the value 10 (which is 1 / 0.1). To focus at infinity, pass in 0 (informally, the value of 1 / infinity). You can pass in non-integer values too. For example:

```
rpicam-still --lens-position 0 -o infinity.jpg
```

...will set the focus position to infinity and not move the lens again.

High Dynamic Range imaging

The Camera Module 3 supports High Dynamic Range (or HDR) imaging. To use it, specify the `--hdr` option on the command line, for example:

```
rpicam-still --hdr -o hdr.jpg
```

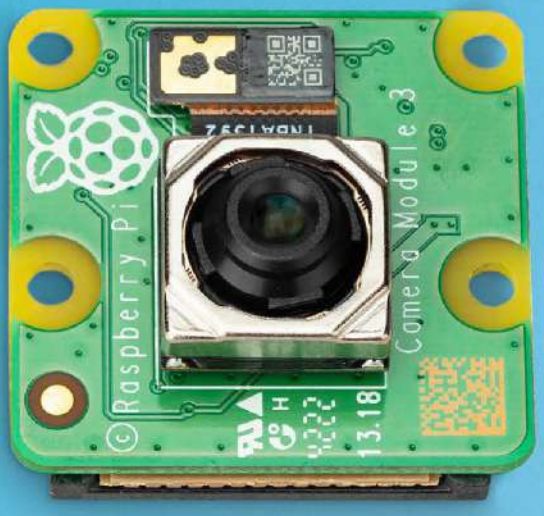
Note that non-HDR captures can be performed at a maximum resolution of 4608×2592 pixels, but HDR captures, because of the special nature of the sensor required to support HDR, are limited to 2304×1296 pixels (exactly half the width and height of the non-HDR mode).

What, no preview?

If you don't see the preview window, please check that:

- You are directly connected to your Raspberry Pi with a screen and keyboard
- You are running the official Raspberry Pi software and that it is fully up to date. If you have made any changes to your Raspberry Pi's configuration please consider reverting back to the standard unaltered Raspberry Pi OS (by re-writing your micro SD card if necessary) and trying again
- All your cables are the right way round and seated correctly
- You are using an official Raspberry Pi camera. Non-official cameras are supported, but may need some extra configuration which the module vendor would have to supply
- You have a good quality power supply. Raspberry Pi power supplies are recommended because they are known to supply sufficient power to Raspberry Pi and the camera

Please check out Raspberry Pi's Camera Forum for more tips and advice if you're having trouble (**magpi.cc/camforum**).



Incredible vision projects with Raspberry Pi Camera

Raspberry Pi Camera Modules, and the fancy new AI Kit HAT+, up the ante for vision-based projects, explains **Rosie Hattersley**

> One of the most appealing aspects of Raspberry Pi is just how adaptable it is

Who doesn't love taking photos? Pets, people, places, events, it's wonderful to be able to take a few snaps and keep the resulting images as a memory you can draw upon whenever you wish to.

Raspberry Pi has some great options for taking both still photos and video thanks to the dedicated Camera Modules that can be attached directly to the board via the CSI (camera serial interface).

Raspberry Pi Camera Modules (**magpi.cc/camera**) cost from just £24/$25 for Camera Module 3, or £48/$50 for HQ Camera (**magpi.cc/hqcamera**).

Of course, Raspberry Pi's compact size means it's ideal for a whole range of visual projects, from a static webcam for video chat from the comfort of your home to outdoor adventures spotting and recording wildlife or natural phenomena in the night sky. You could even add a Camera Module to a robot to act as its eyes and help decide whether to tackle or avoid approaching obstacles.

One of the most appealing aspects of Raspberry Pi is just how adaptable it is: you need only fit the right hardware, often as a HAT to turn it into a specialist device for monitoring, photographing, playing or recording something, be it wildlife, plants, nosy siblings, instruments you want to master, or places you want to memorialise.

This month we're going to look at how to attach a Camera Module to Raspberry Pi and use the Camera Software to integrate it into your projects.

Read these issues!

MagPi 126

We introduced Camera Module 3 in The MagPi issue #126.

magpi.cc/126

MagPi 144

In this very issue, David Plowman, a Raspberry Pi engineer wrote a guide on how to take photos and videos with the Camera Module 3.

Page 56

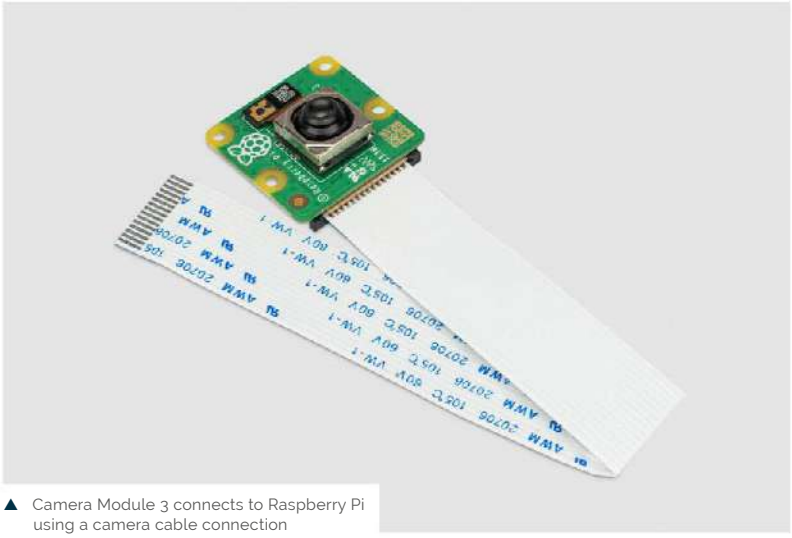
Get to know your cams

Raspberry Pi launched its first Camera Module way back in 2013, with only five megapixels, but an eight-megapixel Camera Module 2 quickly followed in 2016. The latest, Camera Module 3 can now capture HD video at 50fps (frames per second) as well as snagging impressive 12-megapixel stills with HDR (high dynamic range) and autofocus.

There's a Wide version of Camera Module 3, too, for wide-angle photography at up to 120 degrees compared to the standard model's 75 degrees. Both versions also have a NoIR variant that comes without the infrared filter, making the sensor suitable for night-vision use. The NoIR Camera Module is a popular choice for stealthy wildlife photography since many creatures are nocturnal and you don't want to risk disturbing them with unexpected lights.

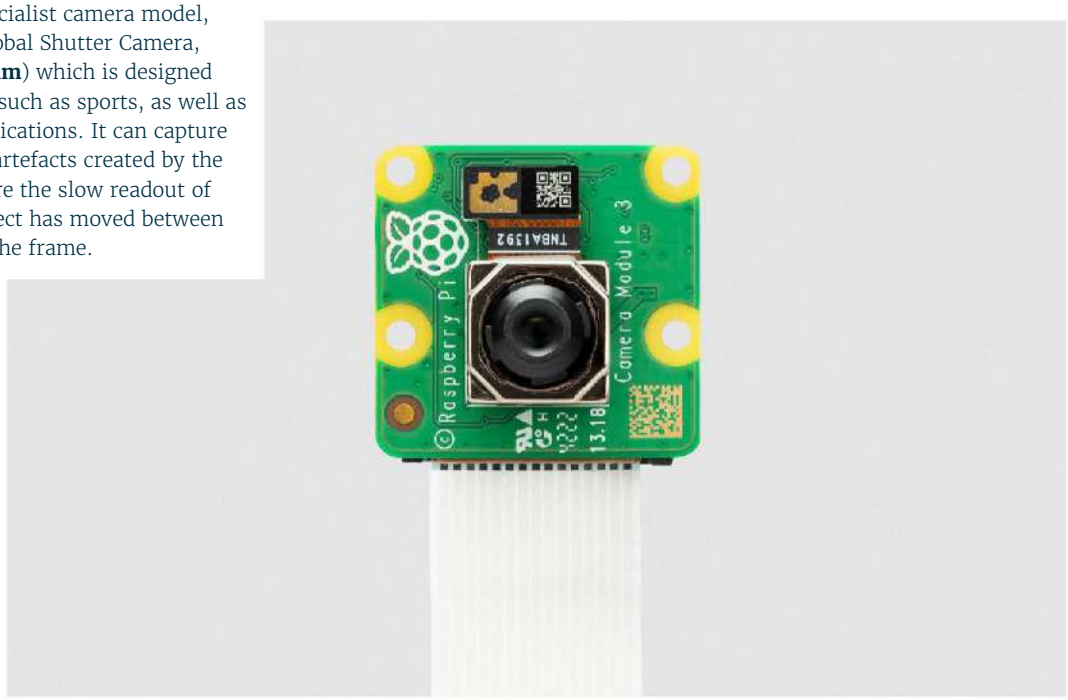
Images and video are typically saved to the same micro SD card that runs Raspberry Pi OS, so it's a good idea to use a card with a plentiful storage capacity (say 16GB or more). Or you can add the new M.2 HAT (**magpi.cc/m2hatplus**) and add a fast NVMe SSD to store large files.

There is also a more specialist camera model, the £/$50 Raspberry Pi Global Shutter Camera, (**magpi.cc/globalshuttercam**) which is designed for capturing action shots such as sports, as well as use in machine vision applications. It can capture rapid motion without the artefacts created by the rolling shutter effect, where the slow readout of the sensor means the subject has moved between the beginning and end of the frame.



▲ Camera Module 3 connects to Raspberry Pi using a camera cable connection

> Camera Module 3 can now capture HD video at 50fps, as well as 12-megapixel stills with HDR



▶ Camera Module 3 features a 12MP sensor with HDR and an autofocus lens

Added intelligence

Raspberry Pi recently launched a brand-new AI Kit (magpi.cc/aikit) that compliments the camera. It that takes advantage of the faster processing speeds of Raspberry Pi 5 and sports a dedicated acceleration processor made by specialists Hailo (**hailo.ai**).

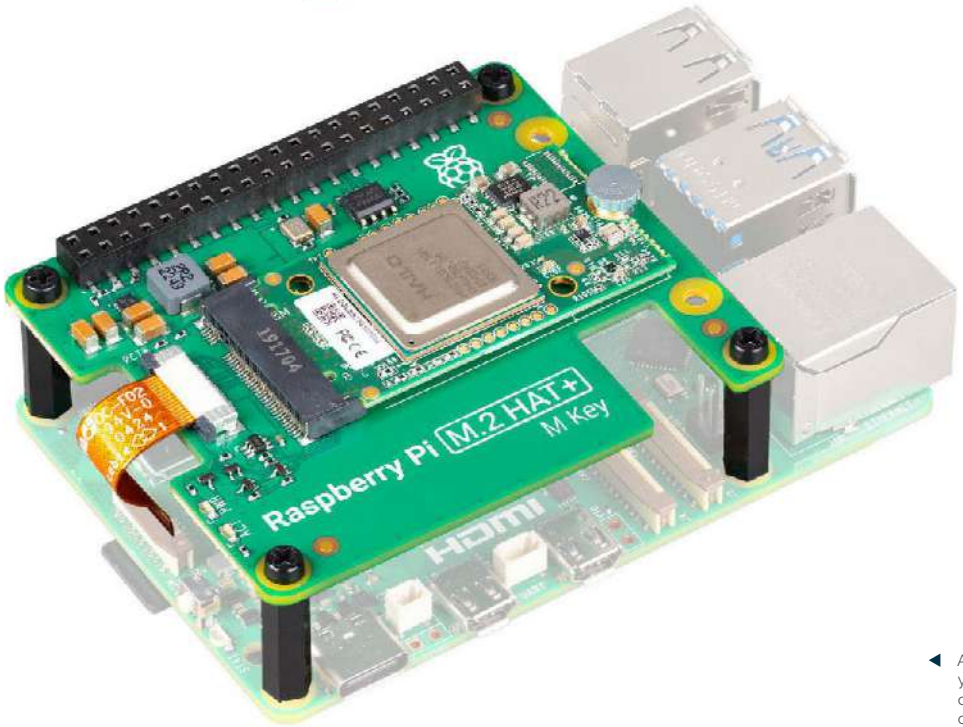
While you've no doubt seen the acronym AI (artificial intelligence) bandied about rather a lot recently, it's more accurate to state that it uses machine learning to work out whether something fits into a particular category or adheres to specific rules.

In the case of Raspberry Pi's AI camera kit, this enables rapid object detection and discerning whether or not an item detected by the Camera Module is, say, a person, a vehicle, a particular type of animal and so on. Developed especially for Raspberry Pi, the Hailo-8L M.2 AI acceleration module provides 13 TOPS of inferencing performance (13 trillion operations per second) and the impressive throughput figures needed for solid neural processing abilities, all for the princely sum of £65/$70.

Wondering what machine learning is all about? Raspberry Pi Foundation has a quick primer (**magpi.cc/machinevision**).

The AI Kit is ideal for fun projects such as wildlife detection, recording perhaps a garden visit from a nocturnal badger but ignoring any cats or foxes that saunter by. And we're sure it'll fit professional applications as well.

> "The AI kit is ideal for fun projects such as wildlife detection"



◀ Adding AI Kit to your camera projects can introduce fast object recognition

Get started with Camera Module

You can use most Raspberry Pi models for photography since all feature a CSI (camera serial interface) for connecting the Camera Module (see 'Compatibility', left).

Power off your Raspberry Pi and use the ribbon cable supplied with Camera Module (**magpi.cc/camcable**) to connect it to the CSI connection on the board. Raspberry Pi Zero and the new Raspberry Pi 5 feature smaller CSI connections, and you will need a special cable to use these devices.

Handily, most resellers include the narrower ribbon cable as part of their Raspberry Pi Zero and Raspberry Pi 5 case kits. You'll need a larger one for other models of Raspberry Pi.

The flex cable inserts into the connector labelled CAMERA on the Raspberry Pi board. Raspberry Pi 5 has two CAM/DISP connectors marked 0 and 1. Connect the camera cable to either of them.

Depending on the model, the camera may come with a small piece of translucent blue plastic film covering the lens. This is only present to protect the lens while it is being mailed to you, and needs to be removed by gently peeling it off.

Locate the Camera Module port and pull up its plastic clip (be gentle). The connectors for the Camera Module and the port need to face each other. Clip them together and carefully push the retaining clip back into place.

James Adams at Raspberry Pi created a setup video for the original Raspberry Pi board, showing how to attach the camera. The process has remained the same since (**magpi.cc/camsetupvid**).

Check that the cable is securely attached to both the Camera Module and Raspberry Pi board, and power on.

Top Tip

Compatibility

Raspberry Pi Camera Modules are compatible with all Raspberry Pi computers with CSI connectors – that is, all models except Raspberry Pi 400 and the 2016 launch version of Pi Zero.

Capture images from the command line

With Camera Module You are now ready to start controlling the Camera Module using the command line.

If you are using Raspberry Pi OS Bookworm (the latest version of Raspberry Pi OS) you will use the new rpicam commands. See 'The rpicam applications', right, for more information.

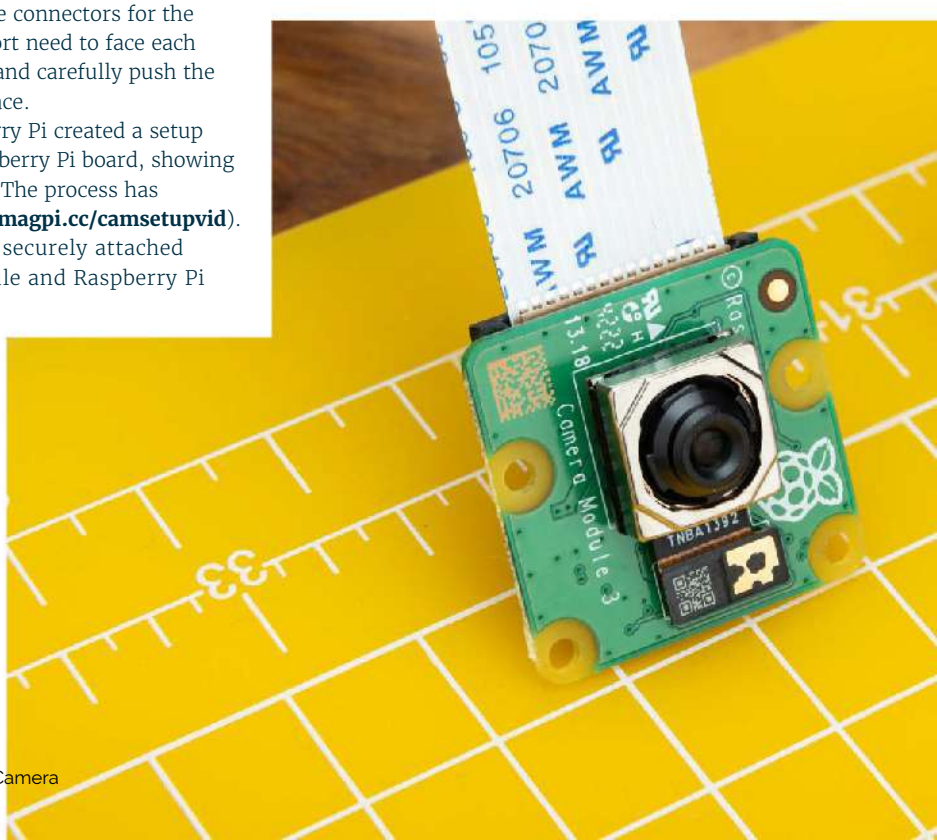
Open a Terminal window (click the black monitor icon in the Taskbar) and type:

```
$ rpicam-hello
```

An image from the camera will appear on the screen. If you get an error message turn Raspberry Pi off and check the cable connections are secure, and that the cables are the correct way around.

Ensuring everything works you can capture an image using rpicam-jpeg and the -output option and a file name.

```
$ rpicam-jpeg --output test.jpg
```



▶ The ribbon cable is used to connect Camera Module to the CSI socket on Raspberry Pi boards

Press Enter to enact the code. You should see a preview window for five seconds. Then, rpicam-jpeg captures a full-resolution JPEG image and saves it.

Experiment with taking screenshots of different sizes and dimensions by adding -height and -width options and entering different numerical values, For example, the following command displays the preview window for 2 seconds, then captures and saves an image with a resolution of 640×480 pixels:

```
rpicam-jpeg --output test.jpg --timeout 2000
--width 640 --height 480
```

See the Raspberry Pi documentation for more information on rpicam and how to use it from Terminal: **magpi.cc/rpicam**.

Use Python to control the camera

The Picamera2 library is a rpicam-based replacement for Picamera, which was a Python interface to Raspberry Pi's legacy camera stack. Picamera2 presents an easy-to-use Python API.

Documentation about Picamera2 is available on GitHub (**magpi.cc/picam2git**) and the Picamera2 manual (**magpi.cc/picam2man**).

Recent Raspberry Pi OS images include Picamera2 with all the GUI (Qt and OpenGL) dependencies. Recent Raspberry Pi OS Lite images include Picamera2 without the GUI dependencies, although preview images can still be displayed using DRM/KMS.

If your image did not include Picamera2, run the following command to install Picamera2 with all of the GUI dependencies:

```
sudo apt install -y python3-picamera2
```

Open Thonny Python IDE from the main Raspberry Pi OS menu, create a new file and save it as camera.py. Note that you should definitely not name it picamera.py as this will overwrite the control app.

Enter the code from camera.py, press Save and Run. Camera Module will display a preview for five seconds, and save the file as test.jpeg in your home folder.

The rpicam applications

Raspberry Pi OS Bookworm renamed the camera capture applications from libcamera to rpicam. Symbolic links allow users to use the old names for now, but you should adopt the new application names as soon as possible. Raspberry Pi OS versions prior to Bookworm still use libcamera name. Here are the rpicam applications you can use from the command line:

- rpicam-hello: A "hello world"-equivalent for cameras, which starts a camera preview stream and displays it on the screen.
- rpicam-jpeg: Runs a preview window, then captures high-resolution still images.
- rpicam-still: Emulates many of the features of the original raspistill application.
- rpicam-vid: Captures video.
- rpicam-raw: Captures raw (unprocessed Bayer) frames directly from the sensor.
- rpicam-detect: Not built by default, but users can build it if they have TensorFlow Lite installed on their Raspberry Pi. Captures JPEG images when certain objects are detected.

Recent versions of Raspberry Pi OS include the five basic rpicam-apps, so you can record images and videos using a camera even on a fresh Raspberry Pi OS installation.

Top Tip

Camera on

The Camera configuration should be enabled by default. You can go to Preferences, Raspberry Pi Configuration and check that Enable Camera is toggled on, then restart your Raspberry Pi.

camera.py

DOWNLOAD THE FULL CODE:
magpi.cc/github

Language: **Python**

```
001.  from picamera2 import Picamera2, Preview
002.  import time
003.
004.  picam2 = Picamera2()
005.  camera_config = picam2.create_preview_configuration()
006.  picam2.configure(camera_config)
007.  picam2.start_preview(Preview.QTGL)
008.  picam2.start()
009.  time.sleep(2)
010.  picam2.capture_file("test.jpg")
```

Camera Module projects

Get started with Camera Module with these great projects

> You will need to choose a discreet location where you have a clear view

Capture wildlife in action

We've covered lots of different wildlife-detecting projects in The MagPi, from the amazing Nestboxlive (**mapgi.cc/birdbox**), which places Raspberry Pi hardware and cameras inside bird boxes in nature reserves in the UK and beyond, to all sorts of self-assembly models that keep the elements out by housing the camera gubbins inside a plastic tub.

If you like the idea of making your own Raspberry Pi camera-based box, you can buy a kit from the likes of Naturebytes (**magpi.cc/naturebytes**). Then follow the instructions to assemble your own using a NoIR Camera Module and any standard-size Raspberry Pi. Along with the other static photography projects covered here, such as the timelapse one, you will need a housing large enough to include a power brick that will keep the Raspberry Pi wildlife camera running.

You will need to choose a discreet location where you have a clear view but the bird box is positioned high enough up that doesn't attract predators (trauma all around!) and the location can be monitored throughout the nesting season without the birds being disturbed. Parents note that this tutorial is part of the National Curriculum too. Read more on *The MagPi*'s website: **magpi.cc/infraredbirdbox**.

Watch something grow with a time-lapse

Time-lapse videos are great for summarising a project build, showing the seasons changing, or something growing. They are one of the best starter projects for Raspberry Pi Camera Module, although you won't see the results for a while.

You can either initiate the timelapse from your Raspberry Pi desktop computer or, more likely, will want to trigger it remotely using SSH to control both Raspberry Pi and Camera Module from afar.

You will need a steady mount to avoid camera wobbles or worse; you don't want to return to the scene of a days' or weeks-long shoot only to find the camera has fallen over leaving you with no usable footage.

Phil King's detailed tutorial in HackSpace magazine (**magpi.cc/timelapse**) walks you through the set up, and recommends mounts such as the ZeroView which allows you to attach your Raspberry Pi Camera Module rig to a window and shoot footage through it. It was created using libcamera, but can easily be updated to rpicam.

Phil strongly recommends creating test shots so you know what's in the frame, and explains how to capture code to preview images if you're triggering the camera remotely. To commence time-lapsing, Compiling the resulting footage is straightforward too and can be done on Raspberry Pi directly.

> Raspberry Pi cameras are ideal for capturing celestial visual spectacles

Ward off unwanted visitors

Parents, nosy neighbours, annoying siblings: who wants them creeping around? Kids keen to try out Camera Module can turn the tables on snoops with a spycam that sends an alert if someone uninvited enters their domain.

The Parent Detector (**magpi.cc/parentdetector**) tutorial is easy to follow and explains how to set up Raspberry Pi Camera Module and PIR motion sensor to trigger video recording should an unexpected visitor be detected. It's clever enough not to trigger if the door simply rattles, has a stealth mode (no telltale red recording light) to fool suspicious types and provides video evidence should a wannabe snoop plead innocence. Should you receive an intruder alert from the Raspberry Pi camera you can view the footage in OMXPlayer to check it wasn't just the cat paying you a visit before you launch in and raise merry hell with the oldies about an invasion of privacy.

Capture celestial wonders

Astrophotography and Raspberry Pi go together very well indeed using Raspberry Pi smarts to track constellations across the night sky, so the telescope is primed for the perfect shot. Projects such as Joe Kutner's autoglider showcase the possibilities to rather impressive effect: AutoGlider (**magpi.cc/autoglider**).

Hubble Pi, meanwhile, makes use of Raspberry Pi's own camera modules to take shots: magpi.cc/hubblepi. With Hubble Pi, Raspberry Pi 4 and the HQ camera are used alongside a Maksutov telescope and established astronomy software including KStars to build a picture of the night sky and identify areas of photographic interest. Maker Santiago particularly liked the HQ camera's "bigger, exposed sensor [that] would allow it to essentially use the telescope as a big mirror lens" and found it good value for the quality it could offer. Santiago created a Python-based interface he called Astrocam that supports touchscreen control and allows him to control the ISO, shutter speed and exposure times as well as capturing multiple shots at once, all of which make it a cut above the average USB camera used for astrophotography.

This year Europe and swathes of the northern hemisphere have been treated to some incredible aurora borealis activity while other places witnessed an eerie solar eclipse. Raspberry Pi cameras are ideal for capturing celestial visual spectacles (**magpi.cc/aurora**) as well as being popular for tracking asteroids and constellations with AllSky Cameras. We were really taken with Svalbard resident Frank Prins' Northern Lights project in which a weatherproofed to -50C Raspberry Pi and Camera Module tucked inside a 3D-printed tube automatically detect the aurora borealis, wake up and record the spectacle for viewing an whatever hour he seems fit.

Because it is not dependent on being sited in a dark sky region (and should work well in light-polluted urban locations too), we like the meteor-detection possibilities of using Camera Module 3 Wide, especially as the setup is clever enough to discern shooting stars and falling meteors, then track and record them as they pass across the night sky (**magpi.cc/meteortracker**).

ArmPi FPV AI Vision

▶ HiWonder ▶ **magpi.cc/armpifpv** ▶ From £200 / $300

A sturdy robotic arm with computer vision for an impressive range of AI abilities. By **Phil King**

SPECS

FEATURES:
6DOF (including gripper), HD camera, breakout board, metal base and fittings

CONNECTIONS:
Breakout board screw terminals for power, JST connections for servos

SERVOS:
LX-255 (base) — 25kg torque; LX-15D — 17kg torque; Anti-blocking bus (gripper) — 8kg torque

DIMENSIONS/WEIGHT
277 × 177 × 428mm; 1.2kg

▶ Equipped with a wide-angle camera, the robotic arm has 6DOF including a strong gripper to pick up objects up to 500g in weight

Verdict

Great value for money considering the sturdiness of the arm, built-in camera, and AI vision possibilities

9/10

Equipped with a camera, the ArmPi FPV is a capable robotic arm that uses AI computer vision to help it perform a range of tasks. The kit is available with or without Raspberry Pi 4 or 5 (4GB or 8GB RAM) included.

There are also three editions to choose from: Standard, Advanced and Developer. The Standard kit features the robotic arm, breakout board (one of two versions, for Raspberry Pi 4 or 5), power supply, paper 'map' (more on that later), coloured blocks, plain blocks, coloured balls, and tags. The Advanced kit adds a couple of flat-pack shelving units for 'warehousing' operations, while the Developer kit adds several extra sensors, a dot matrix display and a fan.

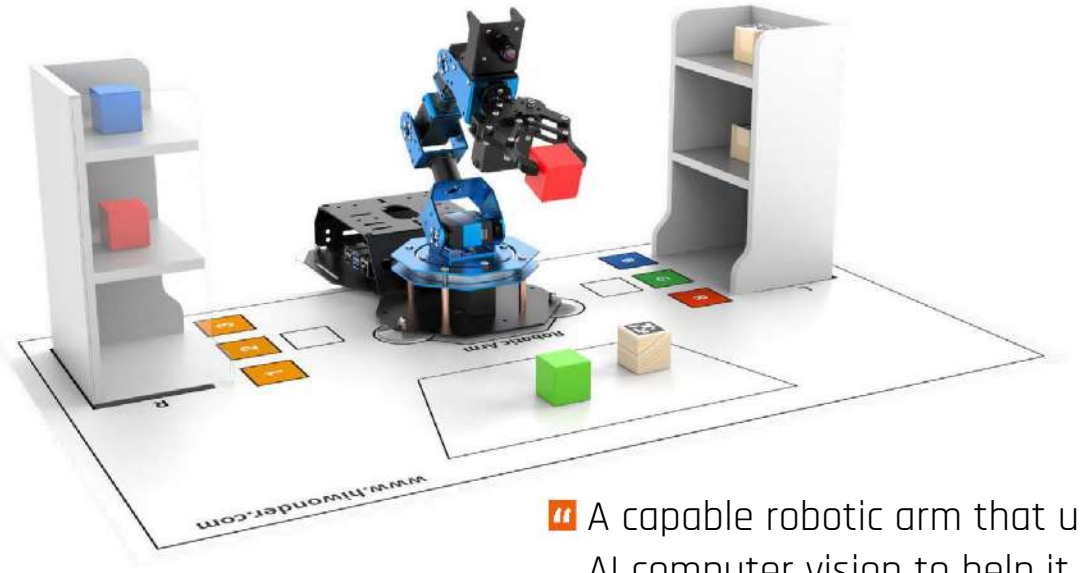
▲ Block-stacking makes use of computer vision and image recognition via Raspberry Pi

Strong arm

We tried out the Advanced kit with Raspberry Pi 5. The arm comes ready-assembled, sitting on a solid black metal base whose rear section holds Raspberry Pi, secured with screws and pillars. Four suction cups stick the base to a desk or the floor – not the most secure of methods – but the unit itself is very sturdy.

The breakout board is mounted on Raspberry Pi and wired from JST connectors to the arm, with some spare servo and SPI sockets. Power – for the arm and Raspberry Pi – is supplied from a 7.5V 6A DC PSU via a barrel adapter with two wires that fit into screw terminals on the breakout board.

The arm itself has six DOF (degrees of freedom) counting the gripper on the end of it, which can be rotated via a 'wrist'. The base servo enables the whole arm to rotate, while three vertical elbow joints, with anodised metal brackets, enable countless possible positions. The servos are of three types/sizes, each with 0.3° accuracy, 0–240° rotation range, and feedback for angle read, temperature, voltage and position. Arm movement is reasonably smooth, and not particularly noisy.



▲ The ArmPi FPV comes with a paper 'map' with block and stacking zones for AI tasks, plus optional shelves for warehousing

> A capable robotic arm that uses AI computer vision to help it perform a range of tasks

A view to a thrill

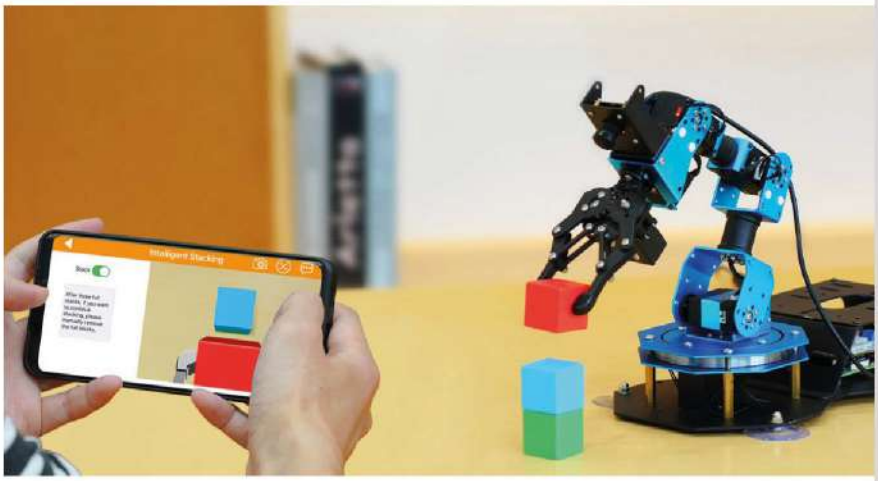
Mounted just behind the gripper, the HD camera offers a wide-angle 120° view. It's connected by a long, loose cable to one of Raspberry Pi's USB ports, with no clips to secure it to the arm, although it does pass through a cutout in the metal base.

The camera is the key to the arm's AI abilities. The simplest way to try these out is with the WonderPi smartphone app (iOS or Android). Once your phone is connected to the arm's automatic Wi-Fi hotspot, you can use the app to control it remotely. The gripper, wrist, and base servos are adjusted by pressing on-screen arrows, while the others are controlled by dragging them on a graphic of the arm.

The app's AI modes are where the fun really starts. Item tracking sees the arm move to track an object of the chosen colour (red, green, or blue) held in your hand. Face recognition is self-explanatory, resulting in a gripper 'wave' to say hello. Item sorting and stacking modes see the arm pick up coloured (or tagged) blocks in turn and either place them on the correct squares on the paper map, or stacked up on a blank square. It's impressive to watch in action and the app gives you an FPV camera view. If you have the optional shelf units, you can get it to do warehousing, delivering, and transferring tasks.

The app is just a showcase for the arm's capabilities, however: a starting point. By following an extensive array of online tutorials, you'll learn how to program it with Python, use OpenCV for image recognition, employ ROS (Robot Operating System) and MoveIt motion planning, use inverse kinematics to reach a desired arm position, and much more.

▼ A smartphone app enables manual remote control and an array of fun AI modes such as block-stacking



TouchBerry Pi Panel PC 10.1

▶ Industrial Shields ▶ **magpi.cc/touchberrypi10** ▶ From £483 / €570 / $614

A robust Raspberry Pi-based touchscreen panel PC for industrial automation. By **Phil King**

SPECS

DISPLAY:
Capacitive touch, 10.1-inch, 1280×800, TFT/IPS, 900 nits, RTD2662 controller chip

CONNECTIONS:
Screw terminals for power, I2C, SPI, RS485, RS232/TTL, 2 × analogue inputs, 3 × digital inputs, 5 × digital outputs

▶ The 10.1-inch version of the TouchBerry Pi has a capacitive touchscreen with a 16:9 aspect ratio and 1280×800 resolution

Designed for use in industrial settings, the newly updated TouchBerry Pi is a panel PC powered by Raspberry Pi 4 that comes in two screen sizes: seven and 10.1-inch – we tested the latter. It has been designed and produced by Industrial Shields, a Spanish company that specialises in industrial automation solutions based on open-source hardware. Its products are used in a wide range of sectors – see **magpi.cc/iscasestudies** for details.

Along with a highly responsive capacitive touchscreen with a 1280×800 resolution, the TouchBerry Pi boasts enough digital and analogue I/O to replace a PLC (programmable logic controller) for some simple automation applications. To this end, it has numerous screw terminals on the left and right sides.

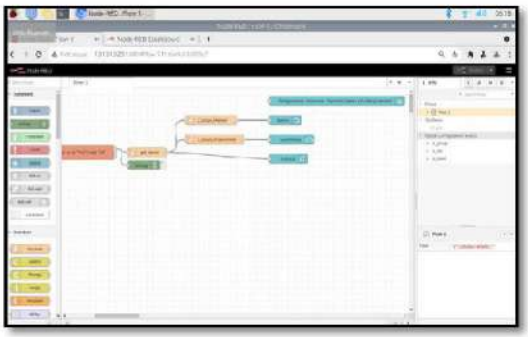
The touchscreen is surrounded by quite a large bezel which forms part of the protective case. With a lot of metal parts, it's a pretty hefty unit that feels really solid – and heavy, at 1.67kg. Six mount points (two top and bottom, one either side) enable it to be mounted using the supplied brackets and bolts.

By default, the case has an IP (ingress protection) rating of 20, for protection against solid objects up to 12mm, but you can pay a little extra (€25) for IP65, which offers full protection against dust and low-pressure jets of water from all directions. Both versions have a wide operating temperature range of 0°C to 50°C, with humidity of 0% to 90%. The unit also has a shock resistance of 80m/s2 in the X, Y, and Z directions.

Verdict

A really solid unit with good hazard protection, a decent touchscreen, and enough I/O for simple applications.

8/10



▲ Node-RED is pre-installed so you can start creating programming flows in a low-code environment within a web browser

> The touchscreen is surrounded by quite a large bezel which forms part of the protective case

▶ The browser-based dashboard for the example temperature sensor Node-RED program has gauges for three measurement units

Powering it `up

After connecting a 12V (2.5A) or 24V (1.25A) DC power supply to the relevant screw terminals on the right-hand side (make sure the polarity is correct!), the TouchBerry comes to life, booting into Raspberry Pi OS – there's no physical power button.

The pre-installed edition of Raspberry Pi OS is the older Buster version, although you could always re-flash or replace the microSD card by opening up the back of the case, a procedure which also enables you to move some jumper switches to reconfigure settings such as voltage levels for the I/O. The system runs fine, although we did experience occasional lag when trying to select items on the desktop.

You can connect to a Wi-Fi network from the desktop or command line (although we needed to manually add a network gateway to gain internet access), or plug in an Ethernet cable for a wired connection. The latter can also be used to network the TouchBerry Pi with a dedicated PLC to add extra I/O and exchange data. Naturally, you can install extra Raspberry Pi software in the usual way.

The only visual clue to the presence of Raspberry Pi 4 is provided by a cutouts for the latter's USB and Ethernet ports, at the base of the unit. So you can plug in a mouse and keyboard if you want. Alternatively, you can install an on-screen keyboard.

Ins and outs

As mentioned, there are two sets of screw terminal connections on either side of the device for I/O. The right-hand side covers all of the main communications protocols: I2C, SPI, RS485 (half or full duplex), and RS232/TTL.

On the left side is the second set of I/Os, which requires an additional power supply to be connected to the 24V COM terminals there. Both the digital and analogue I/Os have self-insulation, so can be connected to power supplies with different voltages than 24V. Two analogue inputs can be used in the default 4–20mA configuration (requiring a 1kΩ resistor) or 0–10V. Three digital inputs and five digital outputs complete the set.

While the total number of I/Os on the TouchBerry Pi is no match for the huge number featured on a dedicated PLC, there should be enough here for many basic applications. We tested it out by connecting a DS18B20 temperature sensor and displaying the data in on-screen gauges using the Node-RED low-code environment, as detailed at **magpi.cc/istempdemo**. A little configuration is required, but it's fairly easy to set up and deploy.

◀ There are I/O (and power) connections – with removable screw terminal blocks – on either side of the unit

LR1302 LoRaWAN HAT + Gateway Module

▶ Elecrow ▶ **magpi.cc/LoRaWANHAT** ▶ From £32 / $41

Improve your IoT devices with this LoRaWAN bundle for full-size Raspberry Pi boards. By **Rob Zwetsloot**

SPECS

CHIPSET
Semtech SX1302 with an RTC battery

SENSITIVITY
-125dBm @125K/SF7, -139dBm @125K/SF12

CHANNELS
8 channels with bandwidth of 125/250/500 kHz

▶ The LoRaWAN HAT in a fetching purple

> The range on the gateway is very good, and we didn't have any problems with devices around our home not reaching it

As the Internet of Things and home automation grow more advanced, the technology that allows devices to connect up around your house (or office, or farm, etc) has also beefed up. We're used to using a lot of wired connections or making use of proprietary wireless formats, however LoRaWAN on Raspberry Pi has opened up whole new avenues to users, with more open and accessible wireless protocols that can cut down on almost all your wires.

The LoRaWAN HAT from Elecrow paired with the firm's Gateway Module is one way to get started with long-range IoT networking on Raspberry Pi, fitting neatly on top of any full-size Raspberry Pi with 40 pins (including Raspberry Pi 1 A+ and B+).

Construction is simple – simply slot the Gateway Module into the HAT's mini PCIe connector, and slot the HAT on top of your Raspberry Pi. There are external antennas to add as well, including a fancy GPS module in case you need to know its location.

From source

The software is a little more involved though. The docs (**magpi.cc/loradocs**) help guide you in the setup, which includes compiling the software from source and connecting everything up to thethings.network so you can control it all remotely. It's a bit of a lengthy process, however it's definitely streamlined once you get past the initial compiling phase. This is not for folks wanting to try out their first IoT set up though, with LoRaWAN products definitely having a more professional user in mind, and the docs don't help you beyond getting the gateway working.

That said, if you know your LoRaWAN stuff, you're all ready to start connecting devices. The range on the gateway is very good, and we didn't have any problems with devices around our home not reaching it. We even put some sensors outside and they worked a treat. The GPS worked well too, something you'd need if you're deploying gateways out on a farm or throughout a wider area. You could probably jury rig it to work as a GPS tracker or Geocacher if you'd like to as well, however there are better ways to do that.

The Gateway Module makes the magic happen

Prosumer uses

This is definitely a very powerful piece of kit, and very reasonably priced at that, although you will need to supply your own Raspberry Pi for it as well. If you're feeling the strain in your home automation network and fancy upgrading to the next level, it's definitely a great way to try out and experiment with LoRaWAN too, and Elecrow also has Node Boards you can use with it if you need to upgrade that part too.

All mounted to Raspberry Pi

Verdict

A great piece of kit designed for folks with serious remote automation needs that fits snugly atop a Raspberry Pi

8/10


FROM THE MAKERS OF The MagPi THE OFFICIAL RASPBERRY PI MAGAZINE
START
NEW 2024 UPDATE
RETRO GAMING WITH RASPBERRY PI
3RD EDITION
180 PAGES OF VIDEO GAME PROJECTS
BUILD AN ARCADE MACHINE
PLAY & CODE GAMES!
File manager
Change color
Change icon
Extra blocks
Save dump
Load dump
Hexeditor
Defragment
Reformat
/ : Select
: OK
: Cancel
Display the list of the files found in the selected VMU.

RETRO GAMING WITH RASPBERRY PI

3RD EDITION

Retro Gaming with Raspberry Pi shows you how to set up Raspberry Pi 5 to play a new generation of classic games. Build your gaming console and full-size arcade cabinet, install emulation software and download original games with our step-by-step guides. You'll discover a vibrant homebrew scene packed with new games for original consoles and legal access to all those retro games you remember!

- *Set up Raspberry Pi for retro gaming*
- *Emulate classic computers and consoles*
- *Learn to code retro-style games*
- *Build a console, handheld, and full-size arcade machine*

BUY ONLINE: **magpi.cc/store**

10 amazing: big builds

Making something big, and making it with Raspberry Pi

Features Ed Rob has been trying to make 'Big Builds' a thing in the magazine for a long time but it never really took off. That hasn't stopped people from around the world putting a tiny Raspberry Pi into projects big enough for a server (sort of). Here are just ten of them.

▲ Raftberry

Floating dock

It can be nice to pootle around a lake, especially with some delicious food and company. This Raspberry Pi-powered raft uses arcade controls to move around on the water.

magpi.cc/raftberry

▲ Teslonda

Custom electric car

Taking a 1981 Honda Accord and souping it up is one thing, then there's making it an electric hot rod. All powered by Raspberry Pi, of course.

magpi.cc/teslonda

▶ Arcade machine

Retro cool

In *Retro Gaming with Raspberry Pi*, we show you how to make your own nifty arcade cabinet powered by Raspberry Pi, and with your own custom vinyls too!

magpi.cc/retrobook3

▼ DoodleBorg

Big rover

PiBorg's biggest robot is a rover they made themselves. It's powerful enough to pull a caravan, which is why it has a tow ball on it. It was built to show just how powerful PiBorg tech is.

magpi.cc/doodleborg

▲ Odyssey Lights

Illuminating Blackpool

One of the biggest Raspberry Pi builds around, these 11-metre-high interactive towers are full of lights, lasers, speakers and other special effects which were made possible with Raspberry Pi.

magpi.cc/odysseylights

▼ Interactive TTRPG table

Digital D&D

Built for in-person Dungeons & Dragons using popular remote virtual table software, the only thing it's missing is an ornate carved fascia.

magpi.cc/dndtable

◀ SailBot

Robotic boat

Tired of winning robotic sailing regattas, a group of university students created an autonomous sailboat that could cross the Atlantic ocean all by itself.

magpi.cc/sailbot

▼ Magic mirror

Rite of passage

A classic project that just about every Raspberry Pi fan has attempted at least once, the software for it is very powerful and easy to use too.

magicmirror.builders

▼ Pinball machine

Steel ball run

The folks at Team Pinball design and build their own pinball machines, and decided that Raspberry Pi was the perfect computer to run these throwback games.

magpi.cc/teampinball

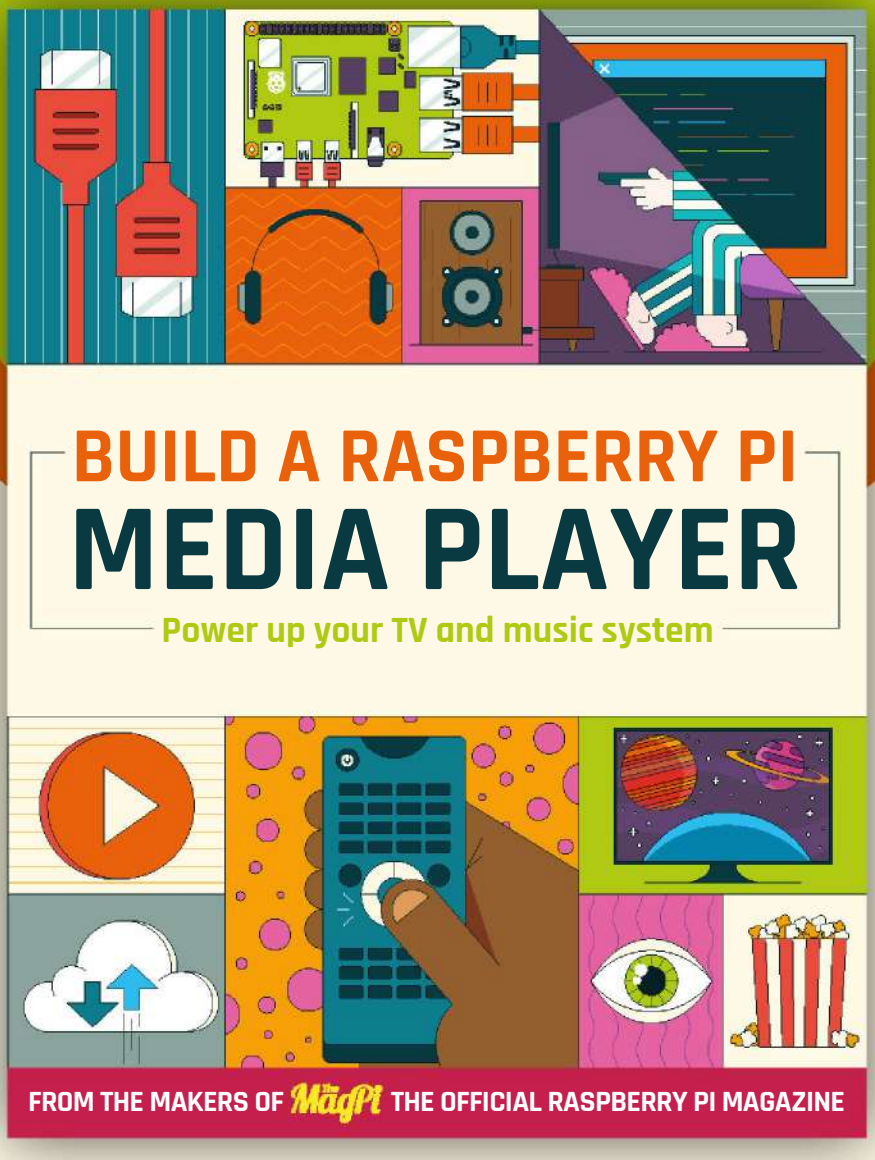
▼ Open Weed Locator

Raspberry Pi farming

This big robot travels fields and uses Raspberry Pi with computer vision software to scan for, and pull up, unwanted weeds and other plants.

magpi.cc/openweed

Your FREE guide to making a smart TV



magpi.cc/mediaplayer

Learn Databasing with Raspberry Pi

Hone your database and SQL skills with these handy resources. By **Phil King**

Introduction to Databases and SQL

AUTHOR

Raspberry Pi Foundation

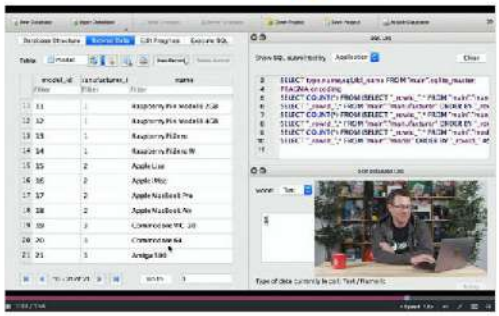
Price:
Free

magpi.cc/rpfdatabases

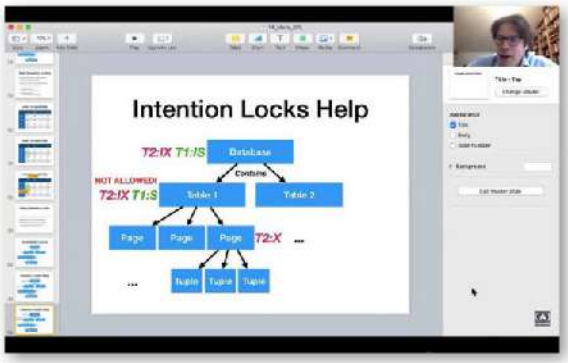
Databases offer a more sophisticated way of storing data than spreadsheets, enabling large amounts of it to be arranged in a logically organised way. This offers the advantages of easy access, data integrity, and security.

Since databases are so useful in a wide range of fields, learning about them is a key part of computer science. Designed by the Raspberry Pi Foundation, this excellent introductory course is available for free via the eDX learning platform. It comprises video and text tutorials, along with quizzes and practical exercises.

After introducing the concept of databases, you'll progress to using SQL (Structured Query Language) to retrieve, filter, and modify the data in a database. Practical activities require the use of either **sqliteonline.com** or the DB Browser for SQLite tool (preferred). The final part of the course covers how to modify an existing database, and explores the relationships between tables and how they can be joined.



Web resources



Useful online resources to gather data

DATABASE SYSTEMS

This 25-hour, two-part video course from Cornell University is a great introduction, covering SQL, storing/indexing data, transaction processing, and more.

▶ magpi.cc/cornelldb

SQL TUTORIAL

This excellent W3Schools resource details all the functions of SQL, guides you through creating a database, and includes useful examples.

▶ w3schools.com/sql

SQLBOLT

A series of 18 interactive lessons, including exercises, to help you learn SQL quickly in your browser. There are some extra topics, too.

▶ sqlbolt.com

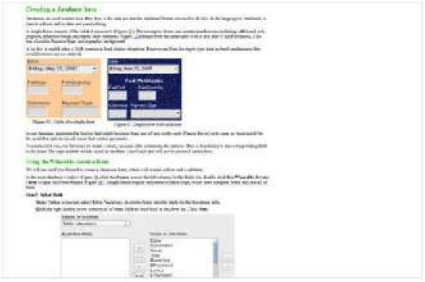
LibreOffice Base Documentation

AUTHOR

The Document Foundation

Price:
Free

magpi.cc/lobasedoc



As well as being free, open source, and available in Raspberry Pi OS, LibreOffice – unlike many other office/productivity suites – comes complete with its own relational database application, Base. While based on the HSQL Database Engine, it's compatible with files in common database formats such as Microsoft Access and MySQL/MariaDB.

It's a good starting point for exploring how databases work, aided by comprehensive official Base documentation that gives you a solid grounding. Planning and creating your first database is made easier with the helpful advice here. It then move on to the basics: creating tables and defining relationships between them, forms and sub-forms for data entry, search queries for extracting specific information, and reports for sharing. All of these can be created using the wizard or Design View. Finally, it covers how data sources can be used with other LibreOffice components.

High Performance MySQL, 4th Edition

AUTHOR

Silvia Botros, Jeremy Tinley

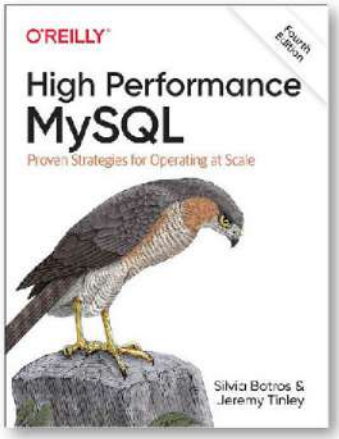
Price:
£53 / $66

magpi.cc/hpmysql

MySQL is an RDBMS (relational database management system) that employs the widely used SQL programming language for interacting with data. It also forms a part of the classic LAMP stack for website design (Linux, Apache, MySQL, PHP).

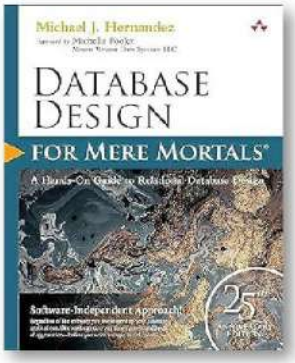
Aimed at readers who already have some familiarity with using relational databases, this in-depth guide promises to take your skills to the next level to unleash MySQL's full power. Over 375 pages, it explores advanced techniques for optimising MySQL databases for improved performance, and how to turn traditional database management tasks into automated processes. Topics covered include schema design and management, indexing strategies, query performance optimization, replication, backup/recovery, and scalability.

The 4th Edition has been updated to reflect recent advances in cloud- and self-hosted MySQL, InnoDB performance, and new features and tools.



Further reading

Books to learn more about databasing



DATABASE DESIGN FOR MERE MORTALS

Covering all the fundamentals in a software-independent way, this classic hands-on guide is a great way to learn database design.

▶ magpi.cc/ddmeremortals4e

SQL ANTIPATTERNS

Antipatterns are erroneous practices that result in poor results. Aimed at experienced SQL users, this book shows you how to fix them.

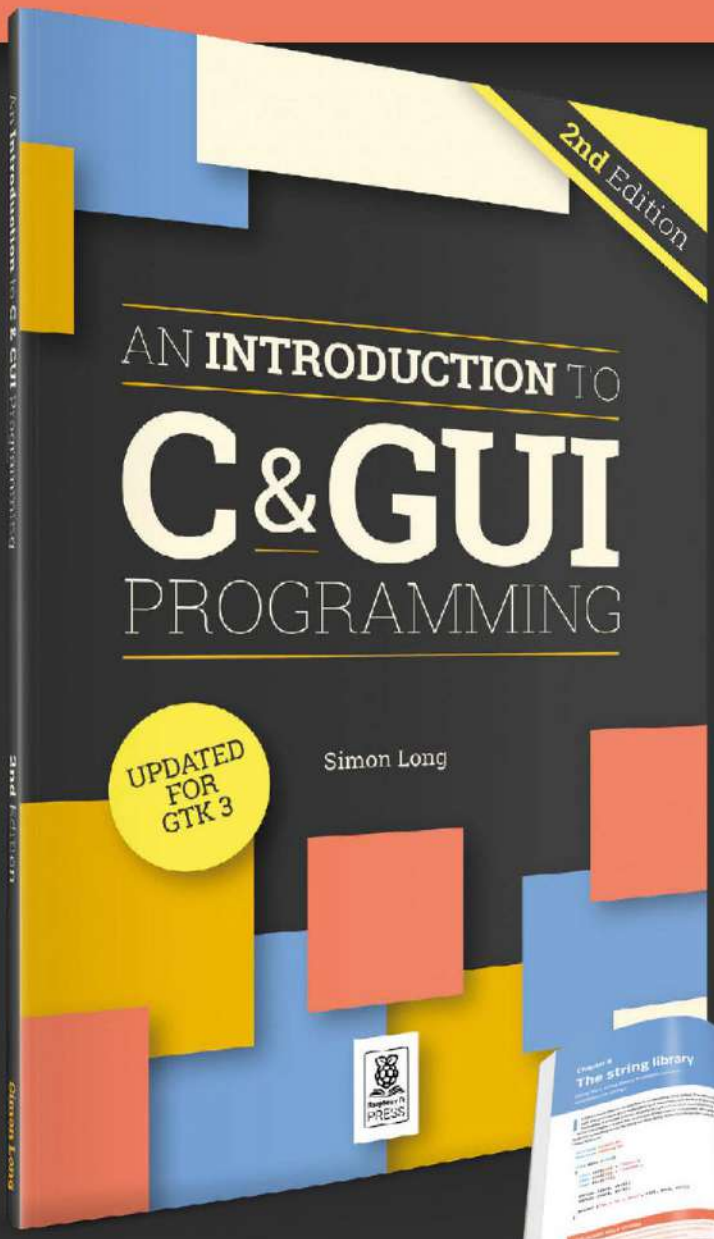
▶ magpi.cc/sqlantipatterns

PRACTICAL SQL

Billed as a 'beginner's guide to storytelling with data', it takes you through all the fundamentals of SQL with plenty of real-world examples.

▶ magpi.cc/practicalsql

AN INTRODUCTION TO C&GUI PROGRAMMING



All you need to know to write simple programs in C and start creating GUIs

Inside:

- Create simple command-line C programs
- Control flow with conditions and loops
- Handle variables, strings, and files
- Design graphical user interface applications in C
- Handle user input with buttons and menus
- Use advanced UI features such as data stores and dialogs

£10 with FREE worldwide delivery

Buy online: magpi.cc/cgui

John Sheehan

The brains behind a California Raspberry Pi meetup that's been running for 11 years

> Name **John Sheehan** | > Occupation **Freelance software developer**
> Community role **Raspberry Jam organiser** | > URL **dev.to/jennasys**

Every month, without fail, we put together the list of upcoming events in The MagPi's community events pages (see p92) and see that there's a Riverside Raspberry Pi Meetup. It's been going for over ten years now and is located in Riverside, a city/county not too far from Los Angeles.

"Someone else actually started the group in 2013," John Sheehan tells us. He's currently the main organiser for the event, which runs monthly. "At the time, I had just started up my own Python meetup group and I was only one of a few dozen members in the Raspberry Pi group. The organiser of the Raspberry Pi meetup that started it ended up stepping down in 2014, and Meetup.com sent out an email saying the group would be shut down if no one else stepped up as organiser. I clicked on a link in the email to find out more about it, and somehow was greeted with a message saying "Congratulations, you are the new organiser!". I'm really not sure what happened there, as it wasn't my intent, but I just went with it. It has now been over a decade and I'm still here."

What is your history with being a maker?

As a kid, I was always taking things apart to see how they worked. Most of those things even got put back together. Taking after my older brother, I started tinkering with electronics when I was a teen. Continuing to follow in my brother's footsteps, I ended up with an undergraduate degree in computer engineering. I got back into electronics during the start of the maker movement when I first learned about the Arduino. I also got heavily involved with a local makerspace for several years that expanded my interest in the process of making, learning the tools, and collaborating with other people.

▲ Showing off the projects and promoting the group at a Maker Faire

◀ The event brings together people off all ages and backgrounds

What is Riverside Raspberry Pi Meetup?
The meetup is an informal monthly meetup for bringing people together that are interested in electronics and embedded software development. But, despite the group name, it is not only for the Raspberry Pi platform. It is as much of a social group as it is a tech group. I try to have formal tech presentations every once in a while, but we more frequently just have open discussions about what everyone is working on, the problems they are facing with their projects, or just sharing the latest news about new maker tech. I do encourage members to contribute talks for the group whenever they can.

What kind of attendees do you get?
Many people who attend the meetup are mostly either new to the Raspberry Pi platform, or are just starting to learn about electronics. But we also have regulars who have been working with or tinkering with embedded platforms for years. Ages range from high schoolers to retirees. Most have at least a little bit of experience in some kind of programming, but not always. The membership is pretty diverse in terms of experience and goals, but we are very open to newcomers.

What are some of your favourite moments from the meetups?
Really, any time someone brings in a project to show off that they have been working on is my favourite part of belonging to the group. I don't even care if it is someone's first project where they just have a blinking LED on a breadboard. I want to hear about it and share in their excitement because I remember being there myself. My reason for having the group is to share what I know and to learn from others. I believe that inspiration is contagious.

> "I want to share in their excitement because I remember being there myself"

Any future events planned?
We meet on the second Monday of every month in Riverside, CA, US. We have snacks. We also try to have a table at local events a few times a year and have members show off their projects, though that schedule varies. We collaborated with a few local school districts several years ago to put on a full Raspberry Pi Jam, and I'd like to do that again sometime as well.

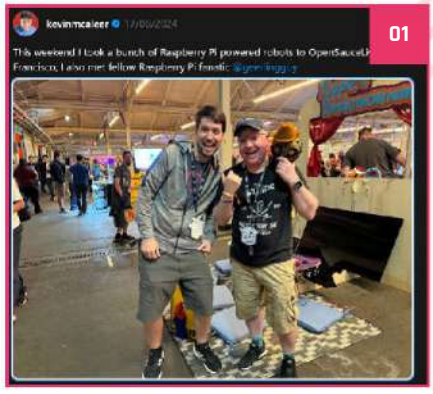
MagPi Monday

Amazing projects direct from social media!

Every Monday we ask the question: have you made something with a Raspberry Pi over the weekend. Every Monday, our followers send us amazing photos and videos of the things they've made.

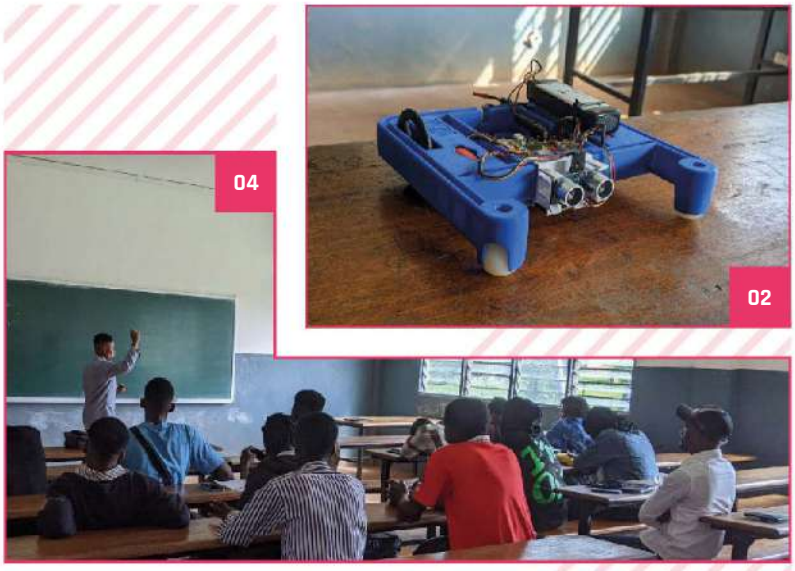
Here's a selection of some of the awesome things we got sent this month. Remember to follow along at the hashtag #MagPiMonday!

01. Two very cool tech YouTubers meeting at Open Sauce, along with Bubo-2T, the tooting owl
02. A very cool glowing lamp – but does that colour-scheme mean it's a Creeper?
03. How big is the train that chuffs past that larger shed, we wonder?
04. Check out **diyelectromusic.com** for more info on this very cool MIDI board
05. Sometimes you build something so good you have to make a second version



Events in pictures: **Raspberry Pi Day 2024 Cameroon**

Community and official events in the wild



This event in Cameroon was held in April. Organisers Mbah Mattklaus and Loic Dessap, brought several people together to show off their Raspberry Pi projects, and to learn about Raspberry Pi in general.

01. Some of the Raspberry Pi goodies available at the event
02. A robot project shown off at the Raspberry Pi Day 2024 event
03. The event drew a fair few young people to learn about Raspberry Pi
04. Giving a talk about their project using Raspberry Pi

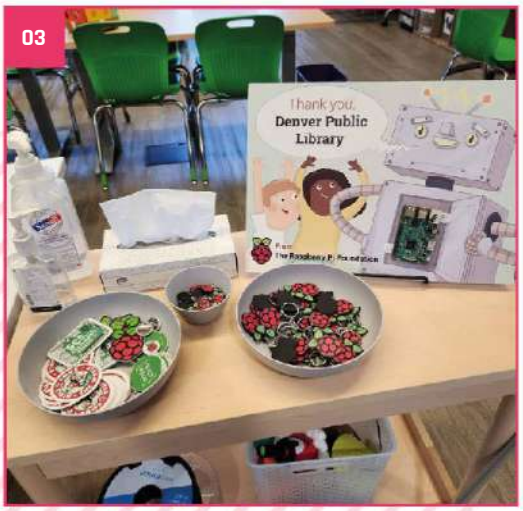
More events in pictures: **Pi Day Raspberry Jam**

Community and official events in the wild

01

After having to cancel due to snow on the actual Pi Day (14 March), the Denver Raspberry Jam folks were able to finally hold their Pi Day celebration on 11 April. Held at the Sam Gary Branch Library ideaLAB, the event saw a huge number of cool Raspberry Pi kits and projects on display for people to see and play with. They even had pie for folks to eat – sounds like our kind of occasion.


02



03

01. A few weeks late, however the Pi Day spirit of the Raspberry Jam was maintained for the event

02. You can make a lot of cool add-ons for flight sims using open source software and a Raspberry Pi

03. Free goodies and hand sanitiser, always the start of a great event

04. People were soon gathering around the displays and checking out the projects

05. Real pies were available for those that wished to partake a slice

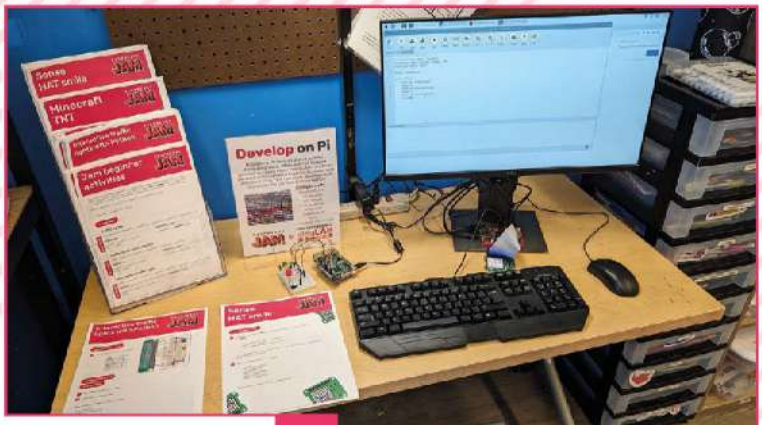
06. Setting up the tables ready for guests

07. This 3D-printed Barad-dûr from The Lord of the Rings has a flaming Eye of Sauron powered by Raspberry Pi

08. Try out Raspberry Pi projects with these dedicated activity leaflets

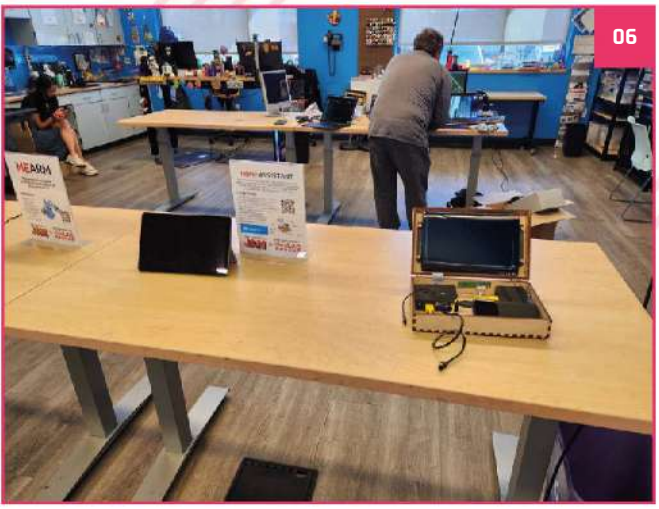
04

05



08

FIND OUT ABOUT NEXT MONTH'S EVENTS ON PAGE 92



06

07

Your Letters

Raspberry Park

I was thinking about the many uses of Raspberry Pi and I was wondering – do you know of any rides in theme parks that use Raspberry Pi? I'm sure some must, maybe they're in that Star Wars Land somewhere.

Beth via email

Unfortunately we've never had anyone tell or admit to us that they're using a Raspberry Pi in their theme park – although we have seen waiting time displays malfunction and turn out to be driven by Raspberry Pi!

We have heard on the grapevine that they do possibly use them at one of the more major, secretive theme park operators but despite our best efforts we've never got them to spill the beans on how Raspberry Pi powers their magic. Maybe because this would unfortunately spoil that magic.

Perhaps we need to find an orange grove and strongarm Eben into opening a Raspberry Pi amusement park there.

▼ User MrDarSwag found a Raspberry Pi at the Jurassic World ride in Universal Studios Hollywood

▶ Pi-Hole is a great way to keep malicious ads off your network

Future forecast

Hi, my dad gets tons of ads watching YouTube on his TV and I want to learn how to use Pi-Hole to block them.

Matthew via email

Pi-Hole is very simple to set up! We covered it in issue 104 of The MagPi (**magpi.cc/104**) with a full set-up tutorial, and there's plenty of info on **pi-hole.net** to help you out as well.

It's very easy to set up and only requires a spare Raspberry Pi for you to do so.

Connectability

I read about the new Raspberry Pi Connect service – will you be doing a tutorial on how best to use it? I'm still not 100% sure how I'd want to use it really.

Kit via Threads

It's incredibly simple to set up so we've not really done much around it yet! However, the team recently added remote shell access and are definitely taking in feedback for more features, so we'll sort out a tutorial on cool stuff you can do with it soon!

Raspberry Pi Connect Beta

▲ Raspberry Pi Connect is very cool and works on even Raspberry Pi 1 computers!

Contact us!

- Mastodon **magpi.cc/mastodon**
- Threads **@themagpimag**
- Facebook **magpi.cc/facebook**
- Email **magpi@raspberrypi.com**
- Online **forums.raspberrypi.com**

Community Events Calendar

Find out what community-organised Raspberry Pi-themed events are happening near you...

01. Getting Started with Raspberry Pi 5: Unlocking Its Potential

Sunday 27 July
Light House Event Center, Ipaja, Nigeria
magpi.cc/vikirp5

This workshop is designed to introduce participants to the basics of Raspberry Pi 5 and explore its diverse capabilities. Whether you're new to Raspberry Pi or looking to enhance your skills, this event will provide hands-on experience and practical knowledge.

02. Melbourne Raspberry Pi Meetup

Sunday 4 August
Docklands Makerspace and Library, Melbourne, Australia
magpi.cc/mrpm144

This meetup is open to everyone with an interest in electronics, robotics, home automation, 3D printing, laser cutting, amateur radio, high-altitude balloons, space tech, etc. Makers are invited to bring along their projects and project ideas, and come connect with other makers. Get your questions answered, show off the work you are doing, and get support to resolve nagging issues.

03. Riverside Raspberry Pi Meetup

Monday 12 August
3600 Lime Street, Riverside, CA, USA
magpi.cc/rrpm144

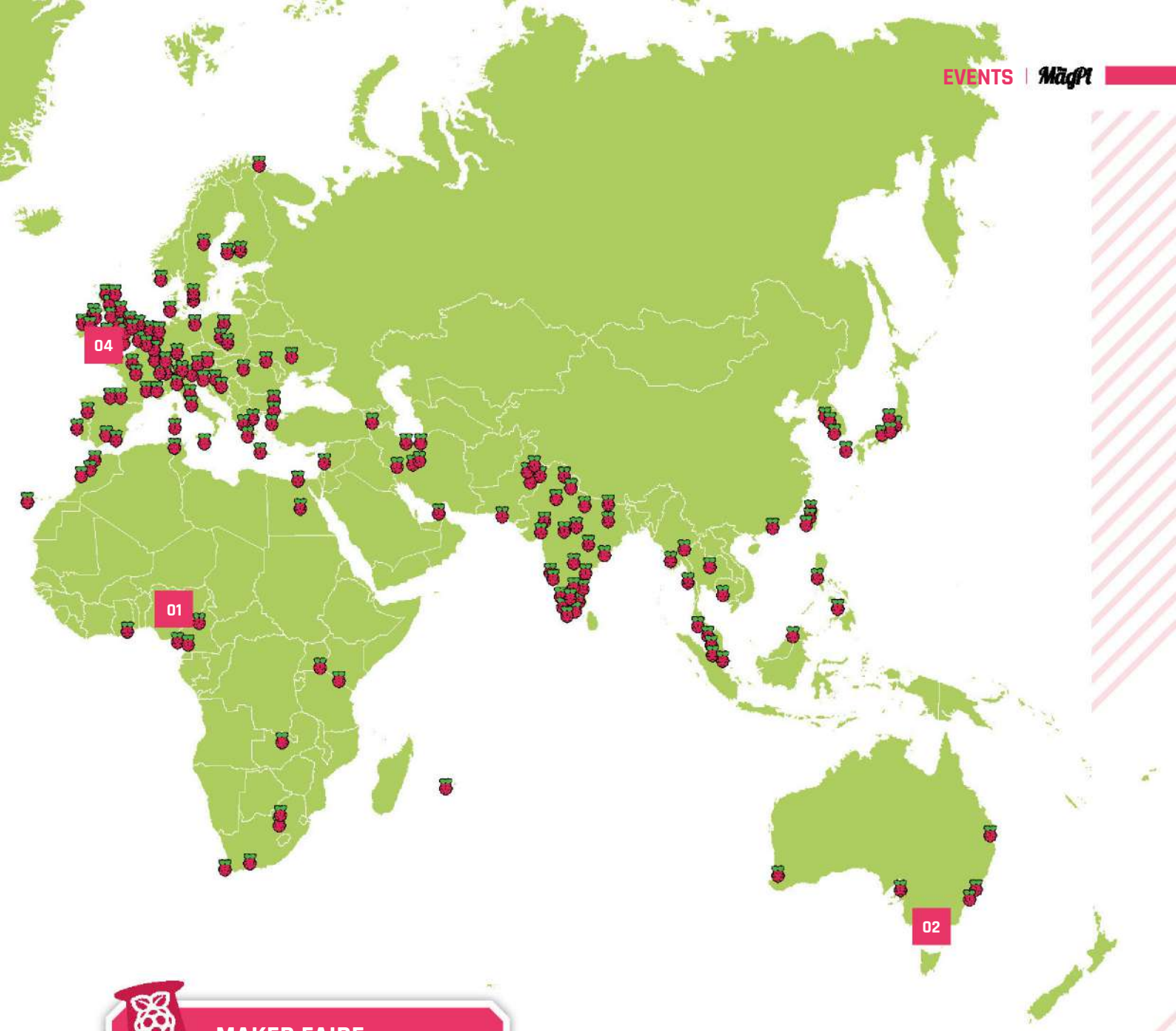
The purpose of Riverside Raspberry is to share knowledge related to Raspberry Pi hardware in particular, and to promote interest in tech development in the Inland Empire in general. The group is currently meeting on the second Monday evening of every month.

04. Tech Cornwall Explorers Summer Sessions Redruth

Monday 12 August and Tuesday 13 August
FibreHub, Redruth, UK
magpi.cc/tcessr24

Two mornings of interactive learning: Three-hour sessions each morning, packed with engaging discussions and hands-on projects. From Basics to Build: start with setting up a Raspberry Pi and progress to constructing your own simple rover. Focus on Teamwork: collaborate, share ideas, and build together in a student-led, inquiry-driven environment.

FULL CALENDAR
Get a full list of upcoming community events here:
magpi.cc/events



Maker Faire Hannover

- Where Hannover Congress Centrum, Hannover, Germany
- When Saturday 17 August and Sunday 18 August

We're proud to partner with Raspberry Pi Approved Reseller BerryBase to be at Maker Faire Hannover for the second year in a row. Come and meet members of the Raspberry Pi team, learn about our latest products, and share what you've made with Raspberry Pi technology.

magpi.cc/mfh24

WIN 1 OF 3 HOME ASSISTANT YELLOW

We reviewed Home Assistant Yellow from the Home Assistant folks in issue 141 and really liked how user-friendly and powerful it was. We now have three Home Assistant Yellow Kits (the version with power supply, without PoE) to give away!

Head here to enter: magpi.cc/win | Learn more: magpi.cc/hayellow

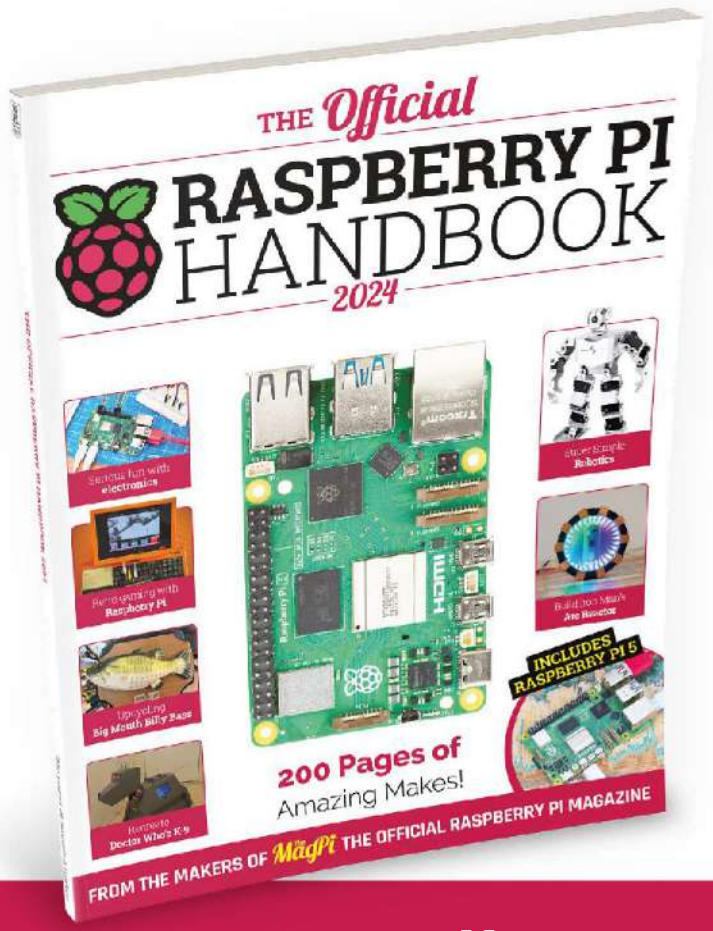
Terms & Conditions

Competition opens on **24 July 2024** and closes on **29 August 2024**. Prize is offered to participants worldwide aged 13 or over, except employees of Raspberry Pi Ltd, the prize supplier, their families, or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from *The MagPi* magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. Winners will be contacted by email to arrange delivery. Any winners who have not responded 60 days after the initial email is sent will have their prize revoked. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram, Facebook, Twitter (X) or any other companies used to promote the service.

THE Official RASPBERRY PI HANDBOOK 2024

200 PAGES OF RASPBERRY PI

- QuickStart guide to setting up your Raspberry Pi computer
- Updated with Raspberry Pi Pico and all the latest kit
- The very best projects built by your Raspberry Pi community
- Discover incredible kit and tutorials for your projects



Buy online: magpi.cc/store

HackSpace

TECHNOLOGY IN YOUR HANDS

THE MAGAZINE FOR THE MODERN MAKER

SUBSCRIBE AND SAVE UP TO 35% on the cover price

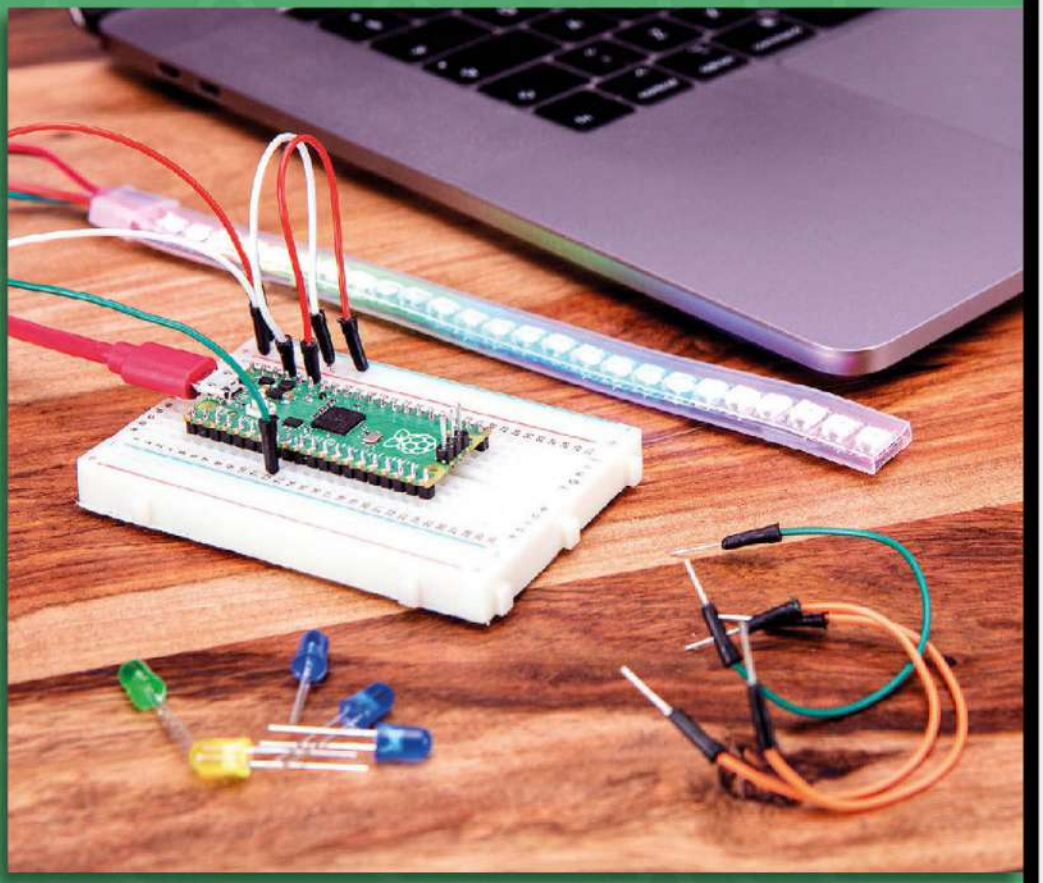
ISSUE **#81**

OUT NOW

hsmag.cc

Available on the App Store

GET IT ON Google Play



Powerful Pico Projects

Make the most of Raspberry Pi's microcontroller

The Magpi **#145**
On sale **29 August**

Plus!

Build a camera that writes poetry

Play a virtual piano with smart gloves

Check for damp with Raspberry Pi

DON'T MISS OUT! **magpi.cc/subscribe**

MASTODON	magpi.cc/mastodon
THREADS	@themagpimag
FACEBOOK	magpi.cc/facebook
EMAIL	magpi@raspberrypi.com
ONLINE	forums.raspberrypi.com

EDITORIAL

Editor
Lucy Hattersley
lucy@raspberrypi.com

Features Editor
Rob Zwetsloot
rob@raspberrypi.com

Sub Editor
Ian Evenden

ADVERTISING

Charlotte Milligan
charlotte.milligan@raspberrypi.com
+44 (0)7725 368887

DESIGN

Head of Design
Jack Willis

Designers
Sara Parodi, Natalie Turner

Illustrator
Sam Alder

Photographer
Brian O'Halloran

CONTRIBUTORS

Nate Contino, David Crookes, PJ Evans, Rosemary Hattersley, Phil King, Chris Lowder, Sean McManus, David Plowman, Ashley Whittaker

PUBLISHING

Publishing Director
Brian Jepson
brian.jepson@raspberrypi.com

Director of Communications
Helen Lynn

CEO
Eben Upton

DISTRIBUTION

Seymour Distribution Ltd
2 East Poultry Ave,
London EC1A 9PT
+44 (0)207 429 4000

SUBSCRIPTIONS

Unit 6 The Enterprise Centre
Kelvin Lane, Manor Royal,
Crawley, West Sussex, RH10 9PE
+44 (0)1293 312193
magpi.cc/subscribe
magpi@subscriptionhelpline.co.uk

This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

The MagPi magazine is published by Raspberry Pi Ltd, 194 Cambridge Science Park, Milton Road, Cambridge, England, CB4 0AB. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products, or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0).
ISSN: 2051-9982.

AI puppetry

On interesting uses for the new Raspberry Pi AI Kit, by **Rob Zwetsloot**

I've probably mentioned here or in past tutorials that I am a streamer when I'm not working on the magazine. I play games and build robot model kits on camera, which probably does not surprise anyone who knows me.

Specifically, I stream as a VTuber. What this means is that I have an animated avatar which I puppet with my face; it uses a mixture of technologies such as face tracking, which is mapped to specific image manipulation software to simulate a living, breathing cartoon character. You also get VTubers who use a 3D model that works similarly to mocap done for movies and video games.

I mainly stream with a 2D cartoon model – one I rigged myself, of course, defining how the different image layers move and warp as I do – however I do have a 3D model which can be controlled with my entire body.

Pose detection

Normally I'm sitting at my PC, so the full-body tracking is not something I use often. I'm able to load it into the popular VRChat software, and at least give the illusion that I am running around with full-body tracking. However, I recently did a very silly stream where I played a Sonic the Hedgehog hoverboard game on the Xbox 360 using the Kinect. It was horrendous, but very funny as I was using a web app full-body tracker with my 3D model.

> "This has got me thinking – can a Raspberry Pi power VTuber tracking?"

Usually, full-body tracking requires motion sensors on specific body parts, however like the Kinect before it, modern software is able to make out your body using machine learning. Just like the new Raspberry Pi AI Kit.

This has got me thinking – can a Raspberry Pi power VTuber tracking? Can it even do the detailed face tracking which you usually need an iPhone to do? I need to start experimenting and hacking I think.

Mobile pose detection?

I've also been dreaming up a mobile puppeteering system. I mean, it's not often that I would need to walk around with a cartoon version of myself on my chest but it could be very funny! And maybe it could be used for a silly costume where you put a screen on your stomach and pretend you're being controlled by Krang from the old TMNT cartoon?

I love it when Raspberry Pi releases new technology that I can get to mess around with and possibly make with stuff with. I really should learn to polish up my projects though so they're a little more presentable. Looking at you, NES controllers with a Raspberry Pi Zero inside that I hacked apart with a Dremel.

AUTHOR

Rob Zwetsloot

Streams online when he's not Features Editor of the magazine, but you very likely have not seen his VTuber alter ego.

magpi.cc

HiPi.io

HIGHPI PRO

The new case from the HiPi.io team

- Rapid tool-free assembly and disassembly
- Large internal volume for HATs
- Compatible with Pi 2/3/4
- Multiple lid options
- Passive & Active Cooling options
- Secure microSD card cover
- VESA mount support
- Molding-configurable output ports customizable for volume orders
- Printed logo for your branding

Available at these great Pi stores:

PiShop.us PiShop.ca PiShop.mx ThePiHut Pi-Shop.ch Welectron.

Contact your favorite Pi store if it's not listed here

PiKVM

Remote control redefined

Manage your servers or PCs remotely!

POWERED BY Raspberry Pi

PiKVM V4 Mini

Small, cost-effective, and powerful!

- Power consumption in idle mode: just 2.67 Watts!
- Transfer your mouse and keyboard actions
- Access to all configuration settings like UEFI/BIOS
- Capture video signal up to 1920x1200@60 Hz
- Take full control of a remote PC's power

PiKVM V4 Plus

The most feature-rich edition

- More connectivity
- Extra storage via internal USB 3.0
- Upgraded powering options
- More physical security features
- Extra HDMI output
- Advanced cooling solution

A cost-effective solution for data-centers, IT departments or remote machines!

Available at the main Raspberry Pi resellers

core ELECTRONICS · PiShop.ca · RPishop.cz · Welectron. · elektorstore · PIMORONI

Pi-Shop.ch · ThePiHut · PiShop.us · CloudFree · KUBII · Cytron Marketplace

ROBU.IN Your ideas, Our Parts · NETTOP COMPUTERS & ELECTRONICS · netsket mall · DIREKTRONIK · RaspberryPi.dk

HiPi.io

No reseller in your country?
Check shop.hipi.io (import fees might apply).

List of official resellers by country: